Volume 9, Issue 5, May – 2024 International Journal of Innovative Science and Research Technology
ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2519

The Role of Machine Learning in Software Development

Dipali B. Tawar
MIT ACSC, Alandi(D), Pune, India

Abstract:- In today's rapidly evolving technological era, the role of machine learning in software development has become increasingly vital and influential. Machine learning has revolutionized various aspects of software development, from code analysis and optimization to prediction and decision-making. Moreover, machine learning algorithms have the potential to significantly enhance the software development process by automating repetitive tasks, improving code quality, and reducing the time and effort required for software testing and debugging. By gaining lots of data and powerful computing resources, machine learning algorithms can be able to analyse patterns and make accurate predictions about software performance, identify potential bugs or security issues, and assist in improving software design and development processes. Furthermore, machine learning can facilitate software maintenance and debugging by detecting anomalies and identifying potential causes of software failures. Albeit, using machine learning techniques into the software development process can greatly improve efficiency, productivity, and overall software quality.

In this paper it is presenting the tools, techniques and the application of Machine Learning (ML) in different phases of Software Development Life Cycle (SDLC) for enhancing and improving the software development process.

I. INTRODUCTION

As Software Engineering has become a vital field and software systems have become more complex due to the performance requirements of their operating environment, the task of designing, developing, maintaining, and adapting these systems has become too difficult for human alone [1]. Therefore, software systems need to be independent and adapt the changing environments to ensure that they should provide expected quality of service. Researchers are increasingly applying ML and AI to fix problems and improve software systems and the software development process. [2]. In recent years, Machine Learning (ML) is gaining in popularity. Machine learning algorithms can assist with software verification and project planning, predicting proof obligations, and identifying necessary team skills. Machine learning can predict software quality using various metrics. One of the popular ML methods for predicting software quality is the CBR (case-based reasoning) method.

Machine learning techniques helps to automate and improve various software engineering tasks, including software testing, maintenance, and management issues. While traditional ML techniques are most commonly used, recent advancements in deep learning techniques are improving ML-based solutions for Software Engineering problems. However, machine learning applications can run into problems with the amount and quality of training data. The most common issues are not having enough labelled data and not having data specific to the context. [3]. To ensure accurate predictions, it's important to clean the data to remove errors and add descriptions before using a machine learning solution.

Machine Learning plays an important role in SDLC and has been used in various domains [6]. For example, these domains include defect prediction, requirements elicitation etc. Therefore, Machine Learning is becoming one of the most significant and important technologies used in SDLC.

Machine learning is used in software development to make software more accurate and reliable. It uses algorithms to find patterns, sort data, and make predictions. This helps identify errors in the code that could cause bugs and other problems.

Now a days Machine Learning has become the most preferred and adaptable technique for developing practical Software Systems in Computer Vision, Speech Recognition, Natural Language Processing, Robot Control, Health Systems, Banking, Defence, E-Commerce and many more environments [9].

Recently, Machine Learning techniques are being used to enhance SDLC and has been used in many areas including behaviour extraction, testing, and bug fixing. [10] This led to a new paradigm in technology invention. During the SDLC, the development team spends a lot of time in discussing how the system must operate, also to decide which features should be prioritized and which ones need to be eliminated. This Requirement Analysis needs to be followed by Design, Implementation, Testing and Integration, and Maintenance phases [11].

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2519

II. RECENT DEPLOYMENTS OF MACHINE LEARNING IN THE SOFTWARE DEVELOPMENT PROCESS

Machine Learning significantly influences the SDLC. This section examines the effects of implementing Machine Learning in each SDLC phase:

A. Software Requirements Analysis
B. Software Architecture Design
C. Software Implementation
D. Software Testing
E. Software Maintenance

A. Software Requirements Analysis and Machine Learning

Whenever we build any system the major phase of software development is software requirements analysis. Without knowing what are the customers need, expectations and requirements from the software we won't be able to build a good system and customer-oriented system.

➢ *Various Problems Arises during this Phase are as Follows:*

- Requirements are incomplete, imprecise
- Requirements are ambiguous
- Requirements are conflicting
- Miscommunication between stakeholders
- Difficult to manage the requirements.

Requirement gathering, a key phase in the Software Development Life Cycle (SDLC), needs the most human involvement.

Researchers have showed a remarkable interest in adopting Supervised ML algorithms like Support Vector Machine, Naïve Bayes, Decision Tree, K-Nearest Neighbor and Random Forest for enhancing Techniques used in Software Requirements. By applying technique like Natural Language Processing (NLP) and Information Retrieval (IR) (Navarro-Almanza et al., 2017) which can lead to more accurate and rapid analysis and reduction in development costs.

B. Software Architecture Design and Machine Learning

The Architecture Design phase (also known as the preliminary or general design stage) is about converting the prescribed requirements after collecting from users from the Analysis phase into an implementable form. Software design has a major impact on software quality. Examples of bad design include anti-patterns, high dependency design, and massive source code files. These things make software engineering tasks harder because they make it more challenging and expensive to fix defects in the software. [12].

The various algorithms used in order to create a flawless design and to enhance different aspects of design like Generative Adversarial Networks (GANs), Decision Trees and Random Forests, Support Vector Machines (SVM), Clustering Algorithms (e.g., K-Means, DBSCAN)etc.

Integrating these machine learning algorithms into the design phase of software engineering can lead to more efficient, innovative, and data-driven design processes, ultimately improving the quality and usability of the final software product.

C. Software Implementation and Machine Learning

Machine learning can assist developers in writing better code by providing context-aware code suggestions and identifying potential optimizations. By analysing existing codebases and learning from patterns, ML algorithms can predict what a developer might want to write next, and this will speed up the coding process and reducing errors which leads to a better quality of software for customers.

Tools like Tabnine and Kite leverage Machine Learning to offer code completions for multiple programming languages. These tools integrate with popular code editors, such as Visual Studio Code and Sublime Text, and learn from coding style to provide personalized suggestions.

D. Software Testing Quality Assurance and Machine Learning

Testing and quality assurance are critical aspects of software development that ensure the reliability and performance of applications. Machine learning can help automate testing processes, making them more efficient and effective. ML algorithms can generate test cases based on code analysis and historical test data, reducing the time and effort required to create comprehensive test suites manually. Additionally, ML can be used to identify potential defects in code, enabling developers to address issues proactively.

Tools like test.ai and Applitools use ML to enhance automated testing capabilities. These tools can analyse application user interfaces, generate test scripts, and detect visual anomalies, ensuring consistent user experiences across devices and platforms.

E. Software Maintenance and Machine Learning

Software Maintenance is the process of looking after a software system after it's been delivered, making changes, and updating it to fix any problems. It's the last phase of the Software Development Life Cycle (SDLC). Maintenance is really important for software quality in Software Engineering. So, being able to predict how it'll go, accurately and quickly, is super important for managing things efficiently in the final phase of the SDLC. [15].

Various Random Forests and Decision Trees, Support Vector Machines, K-Means Clustering by using these machine learning techniques, when applied effectively, can significantly enhance the maintenance phase of the SDLC. They help in proactively identifying issues, optimizing performance, and ensuring the software remains robust and reliable over time.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2519

III. CONCLUSION

Machine Learning is increasingly becoming a pivotal technology within the Software Development Life Cycle (SDLC). ML methods are being applied in complex problem domains with limited human knowledge to create efficient algorithms. ML includes several categories: Supervised Learning, Semi-Supervised Learning, Unsupervised Learning, and Reinforcement Learning.

Our research demonstrates that the Support Vector Machine (SVM) stands out as a prominent supervised machine learning algorithm within the realm of software engineering research. Notably, SVM finds application across all phases of the Software Development Life Cycle (SDLC) due to its ability to address both classification and regression tasks. While SVM may not consistently exhibit superior performance across all scenarios, it remains widely adopted as a fundamental machine learning methodology. Ensuring the accurate application of machine learning algorithms throughout the software development lifecycle poses significant challenges. In conclusion, our study underscores the imperative need for enhanced collaboration between machine learning and software engineering researchers to effectively address the prevailing challenges in this domain.

Also, in order to boost software development productivity, it is imperative for software developers and companies to leverage opportunities within the field of ML. Developers must regularly update their skills and incorporate new ML algorithms and trends into their daily work routines to enhance software engineering practices and streamline production. Companies should encourage employee engagement by offering leadership support and investing in advanced ML algorithms and infrastructure. Today, software developers must possess heightened creativity and intelligence to stay competitive with Machine Learning. ML represents a cutting-edge technology in software engineering, and those who embrace it early on gain a significant competitive advantage.

REFERENCES

[1]. Elhabbash, A., Salama, M., Bahsoon, R., Tino, P. (2019) - Self-awareness in Software Engineering: A Systematic Literature Review – ACM

[2]. Khomh, F., Adams, B., Cheng, J., Fokaefs, M., Antoniol, G. (2018) - Software Engineering for Machine- Learning Applications: The Road Ahead – IEEE.

[3]. Borges, O. T., Lima, M., Couto, J., Gadelha, B., Conte, T., & Prikladnicki, R. (2022). ML@SE: What do we know about how Machine Learning impacts Software Engineering practice? 2022 17th Iberian Conference on Information Systems and Technologies (CISTI).

[4]. Abubakar, H., Obaidat, M. S., Gupta, A., Bhattacharya, P.,Tanwar, S. (2020) - Interplay of Machine Learning and Software Engineering for Quality Estimations – IEEE

[5]. Allamanis, M. (2018) - The adverse effects of code duplication in machine learning models of code Research Gate.

[6]. Alloghani, M., Al-Jumeily, D., Baker, T., Hussain, A., Mustafina, J., Ahmed J. Aljaaf (2020) - An Intelligent Journey to Machine Learning Applications in Component-Based Software Engineering - Springer. Software Maintainability Metrics Prediction – IEEE.

[7]. Aniche, M., Maziero, E., Durelli, R., Durelli, V. H. S. (2020) - The Effectiveness of Supervised Machine Learning Algorithms in Predicting Software Refactoring – IEEE.

[8]. Alsolai, H. (2018) - Predicting Software Maintainability in Object-Oriented Systems Using Ensemble Technique.

[9]. Panichella, S., Ruiz, M. (2020) - Requirements-Collector: Automating Requirements Specification from Elicitation Sessions and User Feedback – IEEE.

[10]. Meinke, K., Bennaceur, A. (2018) - Machine Learning for Software Engineering: Models, Methods, and Applications – IEEE.

[11]. Cetiner, M., Sahingoz, O. K. (2020) - A Comparative Analysis for Machine Learning based Software Defect Prediction Systems – IEEE.

[12]. Zhang, L., Tan, J., et al, D. H. (2017) - From machine learning to deep learning: progress in machine intelligence for rational drug discovery – IEEE.

[13]. Zhang, X., Gu, C., Lin, J (2006) - Support Vector Machines for Anomaly Detection – Research Gate.

[14]. Zhang, X., Zhou, T., Zhu, C. (2017) - An Empirical Study of the Impact of Bad Designs on Defect Proneness -IEEE.

[15]. Gupta, S., Chug, A. (2021) - An Optimized Extreme Learning Machine Algorithm for Improving Software Maintainability Prediction – IEEE.

[16]. Jha, S., Kumar, R., Son, L. H., Abdel-Basset, M., Priyadarshini, I., Sharma, R., Long, H. V. (2019) – Deep Learning Approach for Software Maintainability Metrics Prediction

[17]. Zhu, Y, Chen, L, Zhou, H, Feng, W., Zhu, Q. (2018) – Design and Implementation of WeChat Robot Based on Machine Learning – IEEE.

[18]. Teixeira Borges, O., Colleoni Couto, J., Dubugras a. Ruiz, D., & Prikladnicki, R. (2020). How Machine Learning Has Been Applied in Software Engineering? International Conference on Enterprise Information Systems.

[19]. Zhang, D., & J.P. Tsai, J. (Eds.). (2005). Machine learning applications in software engineering. Google Books.

[20]. Peng, H., Li, B., Ling, H., Hu, W., Xiong, W. and Maybank, S. J. (2017) - Salient object detection via structured matrix decomposition - IEEE.

[21]. Roper, M. (2019) - Using Machine Learning to Classify Test Outcomes – IEEE.

[22].