National Conference on
"Role of Engineers in Nation Building" organized by VIVA Institute of Technology, Mumbai
(6th and 7th March 2020)
International Journal of Engineering Research & Science (IJOER)
ISBN:[978-93-5391-287-1] ISSN: [2395-6992] [Vol-6, Issue-3, March- 2020]

Machine Learning for Software Development

Vishal Sonawane[1], Prof. Pradnya Mhatre[2]

[1]Department of Computer Applications, University of Mumbai
Viva School of MCA, Shirgaon, Virar (East)
Email: vishalsnwn6@gmail.com
[2]Department of ComputerApplications, University of Mumbai
Viva School of MCA, Shirgaon, Virar (East)
Email: pradnyamhatre@gmail.com

Abstract*—Machine learning is the discipline researching pattern for automatically inferring data models. Machinelearning has many applications over life cycle of software development, for testing, to bug fixing and code correction. Howeverbetter understanding of ML methods would help software developers to identify better model for their software applications. Inthis technical analysis we will review and reflect on the applications of ML for software development. In this paper we will lookaround the various algorithms or methods of ML throughout process and their usefulness in Software Development Life-Cycle.We will also discuss some key method and a few obstacles for reaching full potential of ML for software development.*

***Keywords*—Application, Development, Machine learning, Methods, Software.**

I. INTRODUCTION

Machine learning is used for developing application which performs better by experience.ML has advanced drastically in thecourse of decades from research facility interest to common sense innovation in far reaching business use. The use of MLtechniques has been increased since last 5 years. Within artificial intelligence, ML has emerged for developing usefulapplications for computer vision, speech recognition, NLP, robot control, and other applications. In thispaper we will examine about how software development paradigm is changed after introducing machine learning in softwarefield and the ML's impact in software development. By examining the recent literature, we can see small but perhaps significancechanges in SDLC. A far reaching pattern has risen for creating application utilizing ml techniques. Developing and deployingML systems is comparatively fast and non-expensive but maintaining them over a time is not. An ML system has its own specificproblem and additionally software related problem.For instance, probabilistic modeling provides a framework for a machine tolearn from observed data and infer models that can make predict. Uncertainty plays a fundamental role in probabilistic modeling:Observed data can be observed with different models, and thus which model is appropriate given info is uncertain [3].Predictions about future data and the future consequences of actions are uncertain as well. To tackle the ML specific issues in thefield of ML continuous advancement is being made as we speak.

Computer professionals also struggle to operationalize and standardize system software development practices using ML despite these efforts. [3][4]. The one obstacle to tackle is general lack of awareness and the way they could be applied? How does machine learning change software development practices? To systematically explore the ML's impact, we went through the series of interviews given by experts. Experts says "Reinforcement learning (Deep adversarial, Q), semi-supervised and using Closed-loop ML techniques have proven to be beneficial in several stages of SDLC.ML is becoming standardized throughout the SDLC people are researching how to use it to gain insight into where things are going and we see more about deep learning and unique ML methods [1]. According to a 2016 Forrester Research survey, AI also can help in code generation. The survey further revealed that if an AI software is given a business requirement in tongue, it can write the code to implement it — or maybe come up with its own idea and write a program for it[8]. for instance, Microsoft's Intellisense has been integrated with Visual Studio to reinforce the developer experience.

This paper we present the impact of ML technology in SDLC paradigm; also we are going to discuss the limitation for ML indevelopment, and we shade some light on the change in SDLC after introduction of ML tools for software. In this paper we willbe addressing the common questions for Software Developers. We will discuss about recent advances in ML which makingsignificant difference in software industry.

International Journal of Engineering Research & Science (IJOER)
ISBN:[978-93-5391-287-1] ISSN: [2395-6992] [Vol-6, Issue-3, March- 2020]

II. BACKGROUND

The technical briefing is meant for some readers, software engineering researchers, developers and practitioner. The advancement of AI frameworks may be a multifaceted and sophisticated task. Various sorts of procedures of ml improvement are proposed. These procedures share a couple of normal basic advances: setting understanding, information curation, information displaying, and creation and checking. Machine learning is poised to vary the character of software development in fundamental ways, perhaps for the primary time since the invention of FORTRAN and LISP [6].

III. THECONCERNS OF ML

3.1. DATA QUALITY

The most widely recognized issue when utilizing ML is poor data quality. Data is not up to the mark for software development; it needs to be preprocessed. Be able to generate trusted insights with high-quality dataIt is a new area that requires developers to make changes to the way data are obtained and analyzed around the SDLC transition from qualitative to quantitative selenium procedures [11].Selenium was never architected to use ML. Data issues, getting the right data to eradicate bias, taking into account organizationalprinciple. The variance of what the model is trying to do and what you're trying to accomplish.

3.2. BIAS

There can be inherent bias that portion of the training data sets may open on to. Comparably few models can produce more falsepositives and this is true in employing ml in SDLC as well. Selecting the correct algorithms and tools will be an important aspect
of leveraging ml in SDLC[12].

3.3. SKILLS

Skills are a concern. Having the ability and practice to apply ML correctly is key and top of mind for all tech companies today.Another big topic is availability of the upright datasets with correctly labeled data for the development of software[12].

3.4. OTHERCONCERNS

One fundamental concern I have is the understanding of the issue that is attempting to be solved. First there needs to be anunderstanding of whether ml in the SDLC is truly needed. You can do a great deal with essential guideline based methodologiesml can make commotion particularly when you're attempting to accomplish something extremely broad and expansive. I thinkindividuals will in general beginning with something that is pointless excess for what they need. The subsequent issue is building
a one size fits all arrangement. it is extremely difficult to assemble something that can be applied wherever the issue exists inlight of the fact that the setting is constantly significant. Continuously center around quite certain and custom fitted use casesfirst [12].

IV. ML ALGORITHMS

ML manages the problem of building software that enhances their performance at some task through experience. Machinelearning algorithms have been utilized in: (1) data mining issues where large databases may contain implicit regularities that canbe found automatically; (2) ineffectively understood domains where people might not have the necessary skills needed to developpowerful algorithms. (3) Areas where projects should progressively adjust to changing conditions.

4.1DECISION TREE

The concept of decision tree technique is centered on divide-and-conqueralgorithm and it is generally integrated into two indispensable concepts i.e., entropy and information gain. The algorithm in decision tree technique locates the best predictor attributes so that its value is divided according to its information value and thus the highest information gain is indicated by the one with right classification [15].

A decision tree is a logical model that is contributes in research, specifically in decision analysis [8]. Decision tree is a may be a quite tool to return with a decision on the basis of some conditions and their possible consequences. [10]. Decision tree is a method used for classification and regression. Decision tree is a flowchart like tree structure, where each internal node stands for a test on an attribute, each branch expresses an outcome of the test, and each leaf node holds a category. The root node is the

International Journal of Engineering Research & Science (IJOER)
ISBN:[978-93-5391-287-1] ISSN: [2395-6992] [Vol-6, Issue-3, March- 2020]

topmost node in a tree [7]. Decision trees are generated from training data in a top down, general to specific direction. The initial state of tree is root node that is assigned all examples from training the training set. If it is case that each one the examples belong to same class, then no further decision need to be made to partition the examples and therefore solution is complete. If example at this node belongs to two or more classes, then test is made at node that will end in split. The method is recursively repeated for every intermediate node until completely discriminating tree is obtained.M5P is powerful because it implements as maximum amount decision trees as linear regression for predicting a continuous variable. This algorithm is a multivariate tree algorithm which is acceptable for noise removal and also applies for huge database The M5P Introduced by Quinlan, the model tree technique (M5) areoften recognized as an extension to CART.

V. APPLYING ML ALGORITHMS IN SDLC

5.1. BUG FIXING

This is perhaps the biggest areas being reformed with AI innovations. Given the large volume of information that should be tried and human blunder because of ignored bugs, programming testing instruments[1], for example, bugspots give us that projects canuse AI algorithms to auto-right themselves with least intervention of a human developer**.**Different methods are being developed to tackle the problem of device error detection sbp the most popular are machine learning ML techniques in SBP the ml techniques are commonly used to model unstable modules based on historical error data [15].

5.2. CODEOPTIMIZATION

Machine learning requires programming a program model to generate automatic code and answers from production data [16]. The MLalgorithms study and identify the pattern to extract relevant knowledge and make accurate predictions[6] ML is as of now makingcode increasingly efficient. Google's Jeff Dean has said that 500 lines of TensorFlow code in Google Translate is more efficientthan 5000 line of codeAlthough a line of code is not a good metric a decrease of code is tremendous both in programming commitment and in the amount of code to sustain [6]. But what's more noticeable is the manner by this code works: as opposed to alarge portion of a million lines of factual code it's a neural system that has been prepared to interpret.

5.3. CODE COMPLETION

Code completion tool is mostly used in almost all IDE present today. To analyze the progress made so far by the IDE we haveread different papers The most documented usage for ML approaches incode completion has involved the experiments with the n-gram models. Frequency correlation and matching neighbor based approaches are used to boost predictions of ides such as eclipse [2].

5.4. TESTING

For testing entering data in a form field it is recommended to know the 'selector' to find the objects in the HTML page. Whencode and test go out of sync, tests that are irrelevant will fail. Machine learning can be helpful to detect these non-functioningtests automatically and remove them [7].For example, appvance pegged as a software-driven automation testing tool uses ML for output and load testing and produces test cases based on user behavior[5]. Testim.io deploys machine learning to speed upthe execution, and maintenance of automated tests. As one user points out that the tool becomes intelligent when more tests arerun.

5.5 SOFTWARE DEFECT PREDICTION

Prediction of software defects is a very useful tool for evaluating the consistency of produced software product. Predicting defects requires aholistic model rather than a single-issue model that hinges on either size, or complexity, or testing metrics, or process quality dataalone. Bayesian Belief Networks (BBN) proven to be a very useful approach to the software defect prediction problem [13]. A BBNrepresents the joint probability distribution for a set of variables.

VI. MATERIAL AND METHOD

6.1. MATERIAL

The materials we used for our research is articles which presents us the interviews of software personnel and ML personnel hasexperience more than 7 years. We analyze different machine learning papers on Software development for come to theconclusion and answer the commonly asked questions "do the Software engineers/developers need to know the ML practices fordeveloping software?" "What is the impact of machine learning in software industry?" and "what is the future of software

International Journal of Engineering Research & Science (IJOER)
ISBN:[978-93-5391-287-1] ISSN: [2395-6992] [Vol-6, Issue-3, March- 2020]

withML technology and tools? Do they make any difference"? We compared interviews given by ML experts and Non-MLtechnology experts and analyzed their standpoints on topic.

6.2. METHODS

The scope and power to learn interesting models increases each year, because of continuous growth of ML technology.Therefore, what was technical limitation for software industry 5 years ago was, is about to change. This fast growth is reflectingon software developing paradigm. Even so SE community must be aware of and be preparing for adopting the changes inupcoming years. We gathered data from different sources.

VII. CONCLUSION

In this research, we found the recognizable changes in execution among ML and non-ML events. The distinctness lies in avariety of aspects taking account of SE and the context to the programming enhancement. In this paper we discussed how MLalgorithm can be helpful for developing effective software systems. ML algorithm is useful for not only build better softwareapplication but also it can be used to improve the user experience of application. We've seen research which shows that neuralnetworks can write code for new module by inferring old modules. There is continuous progress is being made by the machinelearning community for developers and machine learning practitioner for training and practice purposes. In software sector use ofmachine learning tools is increased over the years since the demand of users for better software experience has risen. That beingsaid, making complex problems manageable remains one of the most important issues for data science. Data engineers are responsible for maintaining the data pipeline that integrates data cleaning capabilities and concept exploration that they are responsible for implementing applications in very complex environments In huger sense this paper represents a step towards software development not as homogeneous bulk but as rich tapestry of varying practices that involves individual of diverse background across non-similar domain. Code reuse is challenging because of different context and input data, and management of data is also a challenge for ML.

REFERENCES

[1] Smith, Tom. "Machine Learning in Software Development - Techniques and Tools - DZone AI." Dzone.com, DZone, 26 Sept. 2019, dzone.com/articles/machine-learning-in-software-development-technique
[2] Contextual Code Completion Using Machine Learning, Subhasis Das, Chinmayee Shah
[3] How does Machine Learning Change Software Development Practices? Zhiyuan Wan, Xin Xia, David Lo and Gail C. Murphy
[4] Applying machine learning algorithm in software development, Du Zhang Department of Computer Science California State University, Sacramento, CA 95819-6021zhangd@ecs.csus.edu
[5] How machine learning a software development paradigm: https://analyticsindiamag.com/how-machine-learning-is-changing-the-software development-paradigm/
[6] What machine learning means for software development: https://www.oreilly.com/radar/what-machine-learning-means-for-softwaredevelopment
[7] SwetaKumari "Comparison and Analysis of Different Software Cost Estimation Methods"(IJACSA) International Journal of Advanced Computer Science and Applications.
[8] JyotiShivhare, "Effectiveness of Feature Selection and Machine Learning Techniques for Software Effort Estimation" June 2014
[9] Machine Learning for software Engineering, Karl Meinke, AmelBennacueer
[10] http://www.qualitymanagementconference.com/effort_estimation.php
[11] https://dzone.com/articles/concerns-using-machine-learning-in-software-develo
[12] Researcher Bias: The Use of Machine Learning in Software Defect Prediction, Martin Shepperd, David Bowes and Tracy Hall
[13] Implementing Decision Tree for Software Development Effort Estimation of Software Project, Sonam Bhatia, Varinder Kaur Attri Student, Dept. of CSE, GNDU, RC, Jalandhar, India Assistant Professor, Dept. of CSE, GNDU, RC, Jalandhar, India
[14] A Decision Tree Model for Software Development Teams Abdul RehmanGilal, Mazni Omar, RuqayaGilal, Ahmed Waqas, SharjeelAfridi, JafreezalJaafar
[15] Software Bug Prediction using Machine Learning Approach AwniHammouri, Mustafa Hammad, Mohammad Alnabhan, Fatima Alsarayrah Information Technology Department Mutah University, Al Karak, Jordan
[16] Optimizing the Code with Machine Learning for Software Development .Sanam Malhotra