International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-9 Issue-2, December, 2019

# Machine Learning: A Software Process Reengineering in Software Development Organization

**Krunal Bhavsar, Vrutik Shah, Samir Gopalan**

***Abstract:*** ***BPR (Business Process Re-engineering) is an organizational mechanism that improves the organizational ability in responding to the challenges of qualitative result by change management and improvement in software engineering processes, productivity, product quality and competitive advantage. BPR inherits, explores and implements the building of process change, to incorporate enhancements to the essential considerations and protocols of (SEM) Software Engineering Management. Machine Learning (ML) can be the key aspect for BPR in software development organization. The goal of this research study is raising the conceptual vision about integration of automation technology like ML and its life cycle development within Software Development Life Cycle (SDLC) of the software product and highlights benefits and drawbacks ML techniques in SPM (Software Project Management), and how to implement ML in standard SEM practices. We have attempted the introduction of machine learning in SEM to determine specific performance and tasks reuse using empirical analysis and discussion on implementation of ML algorithms. The empirical study of software technologies includes control structure of an autonomous software application. In current era, ML imparts consistently promising accuracy in some SEM fields. The goal of this paper is an empirical and analytical study and literature review to propose desired level of quality software, through the comparative evaluation of existing processes and their respective support for Software Quality Engineering (SQE).***

***Keywords: AI - Artificial Intelligence, ML – Machine Learning, SEM – Software Engineering Management, BPR - Business Process Reengineering, SE - Software Engineering, BPM – Business Process Management, SPM – Software Project Management***

## I. INTRODUCTION

Machine learning (ML) is the scientific and analytical study of statistical models and algorithms which is used by the computer system to progressively improve their efficiency and performance on a specific task. ML helps computer system to learn from experience and treat how to do their work on their own when it comes humans and animals nature. ML is emerging as one of the most important technology in the software development industry. ML algorithms adaptively improve performance of the system as the number of samples available for learning, increases. The advancement of technology has been around for several decades, but it is now becoming viable commercially. ML algorithms use machine techniques to “learn” about the info directly from source of knowledge while not depending upon preset formula structure. We are moving into an era where machine learning techniques are essential tools to create value for businesses that want to understand the hidden value of their data. The analytical study in this paper merely focuses machine learning techniques to provide methods for an understanding of processes using data. The business process should be known and several design rules must be followed while designing business control system. The production approaches of the manufacturing industry are not used to design a business process controlling system for a hospital. Because a hospital is not a production or manufacturing industry but specific concepts of production domain such as, what is called a "Product", and elaborated detail about its requirement can be useful. In such kind of scenario, the historical details about the patient information are used to design the business process control mechanism for hospital. A control system may be developed to control the patient information; on the basis of the hospital business characteristics. The business process controlling design and dimensions gains focus on strategic decisions like the reengineering of a business control process. Business Process Management (BPM) is the domain of controlling business processes. The control of a business process has gained its focus widely on business decisions that are made at tactical level of decision making on the real-time and operational (for example budgeting, production planning and resource allocation). This analytical and empirical study about use of technology is an attempt to employ automated business process techniques from the realm of ML to obtain business processes insight. Main challenges for the persons who are responsible for the process and manager, is designing and controlling business management processes effectively and very efficiently. To obtain such type of goal, it is very essential to have a good knowledge and understanding of complete business process.

Revised Manuscript Received on December 30, 2019.
* Correspondence Author
**Krunal Bhavsar,** Research Scholar, Department of Computer Science & Engineering, Indus University, Ahmedabad, India Email: krunalbhavsar@engineer.com; krunalbhavsar.rs@indusuni.ac.in. Contact: +91-9737007007.
Dr. Vrutik Shah, Research Guide, Department of Computer Science & Engineering, Indus University, Ahmedabad, India.
Dr. Samir Gopalan, Research Co-Guide, Business Administration & Management, Indus University, Ahmedabad, India.

Because of substantial amount of information is recorded electronically about business process, the data acquired about business process can be useful to gain a clear vision of the business management process.

Modeling a business process is a very critical task as it requires empirical information, deep knowledge and experience of the process as it is prone to subjectivity and time consuming. To design a system, supporting the business control process, a designer has to construct a business model very accurately describing the process in detail. Some of the information collected at runtime may be used to generate an automated business model automatically, explaining all the business process and events. The topics in this research paper covers discovery related to management trends such as follows, with the help of Machine Learning (ML) techniques.

- Business Process Reengineering (BPR)
- Business Intelligence (BI)
- Business Process Analysis (BPA)
- Continuous Process Improvement (CPI)
- Knowledge Management (KM)

Software engineering and management processes are typically consist of a pool of related tasks known as SDLC such as analysis, designing, implementing, testing, maintaining and reusing software applications. They are very time-consuming, costly and very complex. In addition, the embedded software systems are stepping in almost all kinds of businesses. Kephart [32] concluded that in the very near future, it will be extremely challenging for IT professionals, to manage IT environments. As a result, software engineering field requires highly skilled IT professionals (in software development) to create a powerful software systems, which must be configured installed, and maintained.

## II. RESEARCH OBJECTIVES

Primary objective of the research study is gaining focus on conceptual view for the implementation of trending automation technologies of SE (Software Engineering) for the management of software engineering processes in software development organizations with the help of following analytical and empirical literature reviews:

- Scope of BPR using automation technologies software development organizations
- Focus on trending automation technologies
- Approach ML as a SEM processes

The research highlights an ML (Machine Learning) as the future of automation technologies which can bring revolution in existing reengineering processes. The empirical analysis of SEM processes and approaches in software development organization based on literature reviews, and representation of the conceptualization view using ML life cycle as a part of software system development life cycle stages, is the goal of this research.

## III. BUSINESS PROCESS REENGINEERING (BPR)

BPR is a business management strategy for analysis and redesign of existing business processes within an organization. The discovery of existing business processes is used as an input for business process reengineering. Process discovery is not a system or process redesigning tool although it is useful that in understanding of existing system flow and processes. The goal of process discovery is, identification of what is actually going wrong and where it is required improvements for the system or process. BPR is a very broad area of research and several good methodologies and frameworks have been introduced. BPR is one of the most prevalent development methodologies used across the globe. It allows organizations to make a quantum leap in the nature and level of their services and processes, by making radical changes and eliminating duplicates and irrelevancies, which cause delays.

The literature on BPR reiterates the words of Hammer and Champy [3], "Re-engineering determines first what, a company must do, then how to do it". Incremental change processes, with the aim of gradual improvement, are explicitly rejected as being inadequate in achieving the goals of BPR in that BPR's focus is on radical change - even to the extent that 'what is' is often ignored in favor of approaches which involve radical departures from the status quo in an attempt to concentrate on what 'should be'. The radical discontinuous change approaches which are assumed to be the only effective ones, associated with BPR, are not viable propositions for software development organizations. Such firms are primarily concerned with improving productivity, containing costs (particularly those concerned with rework and maintenance), bringing projects in on time and improving the quality of products produced so as to maintain or increase their position in a highly competitive industry. BPR typically means working in completely different ways with the aim of addressing these issues. However, Craig and Yetton [4] suggest that an alternative approach to managing BPR more effectively might be to begin with what is already in place; there is a learning curve for improving processes that does begin with the prevailing reality - "what is". This view is supported by Davenport [5], who suggests that BPR initiatives fall along a 'business process redesign continuum' ranging from moderate improvement of existing processes to radical redesign of new processes. Krunal et al [15] have concluded that BPR has been enabling organizational capabilities towards implementation of new initiatives with fewer complexities and proposed that a well-structured initial level of PLCF – Process Life Cycle Method suitable to organizational structure requires while attempting to conceptualize integration of ML with PLCF for software development organization.

## IV. MACHINE LEARNING (ML)

Machine learning is a branch of AI (Artificial Intelligence) that enables computer systems to work directly from examples, data, and experience without human interruption. ML is the scientific and analytical study of statistical models and algorithms that is used by the computer system to progressively improve their efficiency and performance on a specific task. Machine learning (ML) enables system models to be trained on data sets before being deployed. Some of the ML models are online and continuously integrate and adapt as new data is ingested. Due to complexity and size, ML patterns and associations could not have been overlooked easily by human observation.

There are seen exciting

advances in machine learning recent years, which have enabled its capabilities across a suite of logical applications. Machine Learning can bring out logical processes and algorithms by learning system data, to enable computer systems to do their tasks on their own very efficiently and intelligently. Availability of required data has enabled ML systems to train themselves by their own. Increasing processing capacity of the computer systems has been supported the self-analytical skills of the systems. The self-learning skill of ML has given greater power to the systems. It has been noticed that system has been performing very well due to this advancement of ML. Machine learning (ML) algorithm finds regular natural patterns from the data that can be helpful to improve business logic insight and helps in making better predictions about process as well help in proper decision making process. ML patters are used day to day activities for making very critical decisions in several business processes like energy load forecasting, stock trading and medical diagnosis. Retailers gain the insight of their customers' characteristics and behavior. Media systems are dependent on ML to sift their business operations in generic user recommended output like list songs or movies etc. for end users. There are following major types of machine learning techniques:

- Supervised Learning (SL)
- Semi-Supervised Learning (SSL)
- Unsupervised Learning (UL)

Some other popular machine learning techniques are: Reinforcement Machine Learning (RML), Regression, Instance-based Learning (IBL), Decision Tree Learning (DTL), Bayesian Learning (BL), Kernel Methods (KM), Clustering Methods (CM), Association Rule Learning (ARL), Artificial Neural Networks (ANN), Deep Learning (DL), Dimensionality Reduction (DR), and Ensemble Methods (EM).

### A. Supervised Learning (SL)

The goal of the supervised machine learning is to create a model that can make an evidence-based prediction in the presence of uncertainty. A supervised learning algorithm uses a known data set as an input and responses output to the data and it trains a model to create reasonable predictions for new data. SL is highly efficient ML technique that trains an ML model label input and output data that are known to the system to make a better future output predictions. Supervised Learning uses classification and regression techniques to obtain predictive result. Classification SL techniques can predict discrete responses. Classification models classify input data into categories. Typical classical model based applications are medical imaging, speech recognition, and credit scoring. Some other for user of this technique are, ensure an email is spam or not etc. Regression techniques of SL help in making prediction about continuous integration and responses such as, fluctuations in demand of power or frequent changes in the scale of performance.

### B. Semi-Supervised Learning (SSL)

Semi-supervised learning techniques help in using of both, unlabeled data for training – as well as labeled data. Semi-supervised learning takes place between supervised learning (with completely labeled training data) and unsupervised learning (without any labeled training data). It uses small level of labeled data while huge level unlabeled data. Many ML (Machine-Learning) researchers have concluded that it can generate considerable improvement in accuracy of learning over unsupervised learning when a small amount of labeled data is used with unlabeled data in conjunction.

### C. Unsupervised Learning (UL)

Unsupervised learning is used to find intrinsic structures or hidden patterns in data. It is used to design inferences from data (sets) consisting of unlabeled input data response. The sample measures typically unknown probability distribution and therefore the learner should produce a basic common model which permits in estimating correct predictions in various circumstances at satisfactory level. Bishop [6] recognized that core objective of a learning is to generalize from its experience. Mohri et al [35] generalized an ability of a machine learning to perform accurately on unseen examples/new tasks in this context, after having experienced a learning data (set). The most common unsupervised learning technique is known as Clustering which is used to find groupings or hidden logical patterns from data for exploratory analysis. Applications for clustering include market research; object recognition and gene sequence analysis. The machine learning theory is a logical observation of the ML (Machine Learning) algorithms and their productivity performance measurement and it's a sub division of theory based science of software engineering. The learning theory usually does not yield any guarantee of the efficient performance of algorithms because the future is uncertain and training sets are finite.

## V. MACHINE LEARNING DEVELOPMENT LIFE CYCLE (MLDLC)

Software planning needs proper communication, requirement elicitation, suitable patterns and phase wise testing of software for fulfilling the client's requirements and completing the umbrella activities. All these can be effectively achieved and predicted by accurate estimates of the budget and project planning.

Core phases of MLDLC (Machine Learning, Development Life Cycle) are as listed below and represented in figure 1:

1. Collect Data
2. Data Refinement
3. Model Training
4. Execute Model
5. Model Refinement

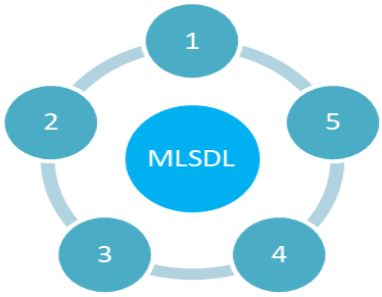


**Fig. 1.Machine Learning Development Life Cycle**

Data collection and labeling includes documentation of data validation and revisit project feasibility. Model exploration includes analysis and selection of proper model suitable to project scope. Model refinement includes optimization of iteration of processes. Testing and evaluation includes validation and analysis of result. And then final model is deployed. In SPM (Software Project Management) a processing on input data and its output plays a vital role, as a project planning phase. Project planning is core SDLC where the MLDLC can be integrated to process on input data of SPM as ML is the king in data processing operations. Other stages of SDLC are focusing monitoring and controlling of output data of planning. Hence ML can play an extreme important role in SDLC management.

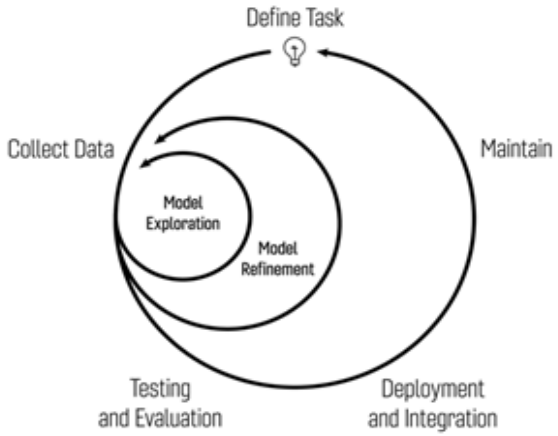


**Fig. 2.Machine Learning Development Life Cycle**

Planning and project setup includes requirement analysis, scope definition and feasibility of project codebase. Figure 2 demonstrates the step for execution of each phase of life cycle.

## VI. MACHINE LEARNING CHALLENGES

Challenges related to proper model and data handling are:

- Pre-processing on data might require specialized tools and knowledge. For example, to select specific features to train an algorithm to detect an object requires specialized image processing knowledge. Different types of data require different types of approaches for pre-processing.
- Data comes in different sizes and shapes. The dataset for real world can be incomplete, messy and in a variety of formats. And so many times a combination of several different data types, such as text, sensor signals and streaming images.
- Choosing the right model is a balancing act. It may take some time to find a proper model to fit the data. Highly flexible models tend to over fit data by modeling minor variations that could be noise. Easy models could assume an excessive amount on the opposite side. Always there will be invariably transactions between accuracy, speed and quality of the model.

ML process workflow always tries to find answers to following three questions:

- What insights are required to be generated?
- What kind of data has been used?
- Where and will those data and insights be applied as an input?

Answers to these questions help you decide whether to use supervised or unsupervised learning for the SEM according input data structure.

## VII. PERCEPTIONS OF SEM

SE (Software Engineering) is the discipline that provides many different ways to choose and adapt different paths to accomplish quality software development. SE gives indirectly indications of learning phases in various directions. Hence, SE is suitable field where we can introduce ML algorithms to develop many more process, product, people and project. Many topics in the Software Engineering [8], [9] can be a fertile soil where many problems and questions can be generated and solved by Machine Learning (ML). The comparative study of some software development tasks using such learning algorithms had been using by many researchers and authors. The purpose of this paper is to have a glance and work on machine learning, in SPM [10] also. The work flow model [11] in SPM gives direction for changing the reports and reporting the problem in SPM. The customer and developer report impart not only the good relation but also give transparencies in the management. SE (Software Engineering) is the protocol or principle which directs adaption of completely different strategy to achieve desired level of quality in software development. Cost Estimation, measurement and analysis of software are the most important factors in SPM [12], [13], [14]. Software project development managers come across software projects that contain inconsistencies or errors and exceed the time limit and budget [30]. A common assumption in the SE is the software development process inherent laws. However, since software product is intangible product, the size and nature of the factors that influence such process are very hard to generate. That is the reason, the software project managers consistently deal with errors or inconsistencies in the software and the budget of the project exceeds it limit along with deadlines. Software managers with organization must decide how to allocate the available resources based on predictions of unknown future. However, the accuracy of development effort prediction can help in reduction of the cost from misleading tendering bids, inaccurate estimation [16] and disabling the monitoring progress. An accurate modeling can assist in scheduling resources and in evaluation of risk factors [17]. In order to develop a project in any engineering domain, the right planning, scheduling and implementing the development process are major factors [36]. For complex or even for simple Project, both planning and scheduling are most important initiatives. The count of potential execution and the way to resource management are crucial and complicated and as well as bulk amount of operations for a tiny size of straightforward projects also. The planning with goals and life cycle model make a path for SDLC (Software Development Life Cycle), SCMP (Software Configuration Management Plan), SQAP (Software Quality Assurance Plan), project plan and schedule and CIP (Component Iteration Plan). Planning and scheduling are clearly different activities [19].

The planning phase [20] is to decide, what must be done and how to do restrictions on the scheduling process. Quality Project Planning (QPP) requires referring the estimation of time and resource for each activity and process. Also, the PM should know and understand the precedence relationships between each umbrella activity or process and other constraints. In order to develop any project, it is necessary to measure the performance of schedule and feasibility of plan. We should use scheduling tools and techniques to optimize project outcomes. To do this, it needs process groups to share the knowledge areas of SPM. These are also called as Project Management Methodologies (PMM) [21]. The Project Management Institute (PMI) recommends eight areas of PMM that are critical to managing a project. The eight areas are management of Scope, Schedule, Time, Cost Management, Quality, Human Resource, Communications, Risk and Procurement. The PMM diagram describes how any software does apply the Project Management Body of Knowledge (PMBK) to be successful in the project [22]. PMM gives also the direction to timely and quality completion of the software project. PPM also reflects SPM Knowledge Areas (SPMKA) [23] and shows not only the different SPM domains but also the scope of project to improve the internal triangle (developed by Lewis) [25]. Another important process of SPM is (RMP) Risk Management Process [26]. Any project RMP places it into four levels.

- Risk Analysis and Identifications
- Risk Monitoring
- Risk Documentations
- Risk Assessment and Feedback

Some of the objectives of software project are based on the RMP measurement. Using the risk objectives and goal of a project, overall performance of the plan and schedule can be accessed. For efficient scheduling, integration of different kinds of data is essential. The model parameters and processes are required to plan a schedule and arrange need of processes, outlined goals and productivity measurement criteria, outlined links between operations and resources, and set of procedure as a rule that should be enforced, in collaboration with the specified system and structured information. Resources include people, machines, and raw materials. Schedules in project are utilized to allocate resources to tasks (or tasks to resources). Core operations and their activities in a project may be deadlines, combination of easy tasks, risks and straightforward actions for development of software feature. In general, objectives for any software project development could be minimizing the total time required for development of project, maximizing the net current project value, or it could be reduced by delay in project release or a product or complexity of software project. Finally, a solution that goes through SPM is more than just plans.

## VIII. EMPIRICAL DISCUSSIONS

The use of Machine Learning is increasing in digital business processes. In 2014, Nadana and Iyakutti [1] proposed an ML based risk control method BSSP (Bayesian Statistics Software Process) and obtained 29% improvement in success ratio of the project by identifying and reducing 33% software risk that resulted in 42% of risk control time compared to tradition software project management frameworks. Technical professionals and project managers must use it as a powerful tool for achieving operational excellence in Software Engineering Management (SEM). SEM is the most important and toughest job in software development organizations. For successful project management, it needs proper planning and scheduling of each software development processes. Machine learning (ML) helps in an automation of software engineering processes and tasks such as system modeling, project planning and development. It's a self-learning approach for a computer system. In this research paper we have focused on an empirical analysis of comparison between how software engineers and machine-learning algorithms and their performance. Comparison between machine learning and software engineering is fair or not? The answer to this question is helps in obtaining the result and insight of the software project management and which process is gives efficient performance. It can also answer to the question, what are processes that can be performed better by human resource rather than machine. And what are those processes that can be done by human resources only and there is no role of machine. Some of the processes that are required coordination of machine for human and vice a versa. And we can also measure what are those processes that give best output by cooperation of both human being and machine learning. The empirical analysis involves the study of the dominance structure of an autonomous software application. We have focused on following dominant steps. Foremost, we have analyzed how we can utilize machine learning to measure the performance and reusability of projects. Then, we have studied and analyzed how software engineers with different domain knowledge levels that they can apply to find out solution to the different tasks. Then, we have tried compared efficiency of software engineers and their reaction against machine-learning algorithms and predicated on criteria such as cooperation of both skills can help in software management processes. Detrimentally, analysis was the comparison between performance indicator level on the basis of empirical and analytical study and nearby conclusion that cooperative work effort gives efficient result.

An ML-based approach improves the reuse of things compared to IoT expert technical engineers. Nascimento [29] et al carried out the same in their empirical analysis; an ML based approach gives better output in comparison to software engineers. Saranya and Sumithra analyzed and argued [28] that SDLC of SE needs changes to improve the quality of large scale software projects and the increasing demand of data processing can be resolved by AI based technologies. Wei and Rana [27] derived that existing SPM tool are management centric only, which very difficult for decision making for project managers. They proposed AI based techniques like ML for improvement in better project planning and scheduling phase of SDLC. In 2019, Krunal et al analyzed [2] that era of AI and ML is the correct time for BPR in software development organization as new technologies have been arising day by day.

They have also evaluated significant rise in AI based technologies, which are in huge demand for transformation of SEM processes where human interference is not essential. Christy and Arunkumar [33] focused financial crisis Prediction based algorithm with capability of early detection of financial crisis possibilities and maximizing performance level and accuracy of system. Rai et al [34] identified several risks like Project planning, testing, technical debt, training, resource management etc. with the agile methodologies and its implementation in software engineering projects, which they reduced with the help of AI based machine learning simulator of MATLAB. In 2019, Gopal and Amirthavalli [37] performed an empirical analysis on case study of open-source software, with an objective to predict relationship between maintainability and metrics, and observed that Instability metric is most influencing amongst all. In 2019, Kumar and Singh proposed [38] a threshold based segmentation using MATLAB tool to automate performance evaluation of students and evaluated that ML based procedure performs more accurately. Ram Kumar et al explored in 2019 [39], about ML based techniques such as supervised, unsupervised and semi-supervised and analyzed that ML integration reduce or eliminates error and automates system processes. In 2019, Bhutada and Morey [40] applied ML (Machine Learning) algorithms on information accrued system and observed that it helps in pre-screening of evaluation of applicant's information. In 2015, Hanchate and Bichkar proposed [41] that Machine Learning (ML) based techniques like classification, case based reasoning, neutral networks, regression trees can improve prediction about the Software Project Management activities like requirements elicitation, planning and scheduling etc. Wan et al [42] performed qualitative and quantitative survey in 2019 and elicited significant difference between ML based system and non-ML based system development in Software Engineering aspects like planning, designing, testing with operational characteristics like differences in skills, problem solving techniques and task reusability, where they found ML based system development is more accurate and robust.

## IX. RESULT AND RECOMMENDATIONS

The era of human driven structure for software engineering management meant defining protocols for deterministic problems solution using a human logic. However it needs a transformation at large scale software developments. Such requirement cannot replace the need of SDLC but demands change in the internal concepts of processing mechanism of SDLC.

### A. Challenges

To modernize organizational capabilities through business intelligence, following are some challenging key findings represented in Table I, that need to be considered for Machine Learning in Software Engineering (SE) practices.

**Table I: Challenging Parameters/Criteria**

| Parameters | Challenges |
|---|---|
| Raw Data | Unavailability of clear raw data. |
| Platform (Technology) | Storage Cluster and Computing platform. |
| Agile Requirement | Changing Requirements |
| Expertise | Technically Skilled Resources |

- Raw Data: It is very difficult to provide a clear raw data for ML to work with its algorithm and logical architectures to interoperate on data based on given platform.
- Platform (Technology): The hearts of the Machine Learning system are storage cluster and computing platform. They will vary on the basis of goal of summation need, learning method and application.
- Agile Requirement: To improve operational business agility and efficiencies, organizations may keep changing requirement and use multiple Machine Learning (ML) frameworks, which lead towards unexpected result.
- Expertise: The team of ML based projects requires a skilled data scientist and analyst as technical product and project managerial member, who produce the desired result that can be used as an input for project planning and scheduling.

### B. Benefits

ML plays the vital role in data refinement where human is not capable to deal with large amount of data directly. ML algorithms have capability of pattern identification, self-learning and training to other data. Table II represents the key benefits of ML in software engineering practices in software development organizations.

**Table II: Benefits of ML in SEM**

| Parameter | Advantages |
|---|---|
| Learning | Self-learning |
| Treat Data | Result oriented data |
| Pattern | Pattern Identification |
| Data Cleanup | Self-cleanup of garbage data |
| System Process | Automation |

- Learning: Machine learning (ML) is a self-learning system with the ability to learn and improve by using algorithms on the basis of input data and results in new insights without external interruption.
- Treat Data: ML algorithms have ability to treat the data using the data. It generates new data using previous operating results on previous data.
- Pattern: ML algorithms can detect data pattern and produce result accordingly.
- Data Cleanup: ML algorithms can prepare their own raw input data by cleaning the pool of garbage data.
- System Process: ML automates the automated system processes to reduce the risk with the project.

### C. Recommendations

With the use of emerging technologies of ML in the software engineering projects, the product and project managers should perform some realty checks for the necessity of ML in the software development. Table III recommends several key parameters that should be consider prior to applying ML in software product development.

**Table III: ML Recommendations for SEM**

| Parameters | Consideration |
|---|---|
| Requirement Analysis | Technical Dependency |
| Algorithm Selection | Feasibility Study |
| Planning & Scheduling | Amendment and Change Request |
| Performance (Time) | Project Timeline |
| System Performance | Operational Logic (System Configuration) |
| Prioritization | Priority of execution of operation. |
| Risk | Risk Identification and Analysis |

- Requirement Analysis: Understand requirement of the project and technical dependencies of ML.
- Algorithm Selection: Ensure that the selected ML algorithms will produce expected result.
- Planning & Scheduling: Any amendment in project requirement should automatically refine project scheduling data.
- Performance (Time): Selection of proper algorithm that can produce necessary result with in available expected time duration
- System Performance: Operational logic for ML based program should overload available memory capacity of the system.
- Prioritization: Understanding and balancing between priority and not priority operations requirements.
- Risk: Don't avoid the risk associated with use of ML as ML has capability to treat the input raw data itself.

## X. CONCLUSION

Change in the software engineering practices is extremely essential nowadays. The increasing demand of data processing in software development process, invites a new text level of processing methods. The big data analysis and data mining operations are existing technologies to compensate the demand for data processing. ML is the domain that process on knowledge to train the data for error free, accurate and rapid application developments. This research paper outlines the challenges and collaborations future of SEM. The empirical result of this research determined by the practical used ML (Machine Learning) techniques such as case based reasoning, classification, neural networks and regression trees, genetic algorithm, rule induction and genetic programming for the subdomain of SEM by researchers and ML software development experts. This research paper widely argues that the Software Engineering can collaborate with domain of AI and ML to produce better software and applications. Managerial and human factors will greatly influence the implementation and acceptance of Business Process Reengineering using Machine Learning tools and techniques.

At last, this research study concludes the conceptual view of scope of enhancement in SEM using empirical studies focusing on integration of ML with SEM and a scope of ML lifecycle in software project development life cycle.

## FUTURE ENHANCEMENT

Machine learning isn't a solitary endeavor; it's a team process that requires data scientists, data engineers, business analysts, and business leaders to collaborate. The power of machine learning requires collaboration so the focus is on solving business problems. The scope future work in Software Engineering Management to extend the proposed experiment of Machine Learning includes:

- Further analytical experiments with different types of machine-learning algorithms;
- Further comparative analysis between machine learning and traditional software development practices into organizations.
- Further empirical analysis of trending software engineering management framework like Agile Scrum [24] and integration of AI based technologies like machine learning to overcome the challenges of Scrum [31], Scrumban [7] and Scrumbanfall [18].

## ACKNOWLEDGEMENT

As a part of my academic affiliation with Indus University for the doctorate degree program in Computer Science and Engineering, through this research study we have attempted to determine the gap between existing software engineering management standard practices and issue with it; and proposing how the ML can help SEM (Software Engineering Management) in BPR (Business Process Reengineering). ML based standard SDLC framework has been demanding to be integrated with software engineering management process life cycles and detail demonstration of integration will be our next goal.

## ABBREVIATIONS

**A**
AI - Artificial Intelligence

**B**
BPR – Business Process Reengineering
BPM – Business Process Management
BI – Business Intelligence
BPA – Business Process Analysis

**C**
CPI – Continuous Process Improvement

**D**
DL – Deep Learning

**M**
ML – Machine Learning
MLDLC - Machine Learning Development Life Cycle

**N**
NLP – Natural Language Processing

**P**
PLCF – Process Life cycle Framework
PPM - Project Management Methodology

**Q**
QPP - Quality Project Planning

**S**

SDLC – Software Development Life Cycle
SE – Software Engineering
SEM – Software Engineering Management
SL - Supervised Learning
SSL - Semi-Supervised Learning
SPI – Software Process Improvement
SPM – Software Project Management
SR – Speech Recognizer
SQE - Software Quality Engineering

**U**

UL – Unsupervised Learning

## REFERENCES


1. S. P. V. Nadana and K. Iyakutti, "Bayesian Statistics Software Approach for Risk Control on Complex Software Projects*", International Journal of Engineering Research & Technology (IJERT), vol. 3, no. 12, pp. 1062-1068, 2014, ISSN: 2278-0181.
2. K. Bhavsar, V. Shah and S. Gopalan, "Business Process Reengineering: A Scope of Automation in Software Project Management using Artificial Intelligence", International Journal of Engineering and Advanced Technology (IJEAT), vol. 9, no. 2, 2019, ISSN: 2249–8958
3. M. Hammer and J. Champy, "Reengineering the Corporation: A Manifesto for Business Revolution", Harper Business Press, New York, 1993.
4. J. F. Craig and P.W. Yetton. "The dual strategic and change role of IT: A critique of business process reengineering", AGSM Working, Paper 94-002 University of New South Wales, 1994.
5. T. H. Davenport, "Process Innovation: Reengineering Work through Information Technology", Harvard Business Press, Boston, 1993.
6. C. M. Bishop, "Pattern Recognition and Machine Learning", Springer, 2006, ISBN 978-0-387-31073-2
7. K. Bhavsar, V. Shah and S. Gopalan, "Scrumban: An Agile Integration of Scrum and Kanban in Software Engineering", International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 9, no. 4, 2020. ISSN: 2278-3075. (In process)
8. R. S. Pressman, "Software Engineering: A Practitioners Approach", McGrawHill, Inc., New York, pp. 976, 1992.
9. I. Sommerville, "Software Engineering", Addison Wesley, New York, pp. 684, 1998.
10. S. Bhatnagar, "Indian Software Industry", Proceedings of the fifth international conference on genetic algorithms, IIM, Ahmedabad, India, pp. 095–124, 2006.
11. M. Chrissis, M. Konrad and S. Shrum, "CMMI: Guidelines for Process Integration and Product Improvement", SEI series in software engineering, Addison-Wesley Professional, pp. 688, 2001.
12. B. Hughes; M. Cotterell and R. Mall, "Software Project Management (SIE)", Tata McGraw-Hill Education Pvt. Ltd., pp. 432, 2011.
13. B. Hughes; M. Cotterell and R. Mall, "Software Project Management (SIE)", Tata McGraw-Hill Education Pvt. Ltd., pp. 384, 2011.
14. P. Jalote, "Software project management in practice", Addison Wesley, New York, pp. 312, 2004.
15. K. Bhavsar, V. Shah and S. Gopalan, "Process Life Cycle Framework: A Conceptual Model and Literature Study of Business Process Re-Engineering for Software Engineering Management", CiiT International Journal of Software Engineering and Technology, vol. 11, no. 6, pp. 096-100, 2019. ISSN: 0974-9748, DOI: SE062019001
16. A. Trendowicz, "Software Cost Estimation, Benchmarking, and Risk Assessment", The Fraunhofer IESE Series on Software and Systems Engineering, Springer-Verlag Berlin Heidelberg, pp. 322, 1998.
17. R. J. Madachy, "Knowledge-Based Risk Assessment and Cost Estimation", Automation of Software Engineering, vol. 2, no. 3, pp. 219–230, 1995.
18. K. Bhavsar, V. Shah and S. Gopalan, "Scrumbanfall: An Agile Integration of Waterfall, Scrum and Kanban in Software Engineering", International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 9, no. 4, 2020. ISSN: 2278-3075 (In process)
19. M. Wall, "A Genetic Algorithm for Resource-Constrained Scheduling", Massachusetts Institute of Technology, Department of Mechanical Engineering, p. 13, 1996.
20. B. Smith and N. M. Crockett, "Business Services: Catching the wind", Bureau of Employment Security, Division of Economic Analysis & Research., p. 20, 1989.
21. D. B. Hanchate and R. S. Bichkar, "Software project contacts by GRGA scheduling and EVM", International Journal of Computer Applications, vol. 97, no. 13, pp. 01–26, 2014.
22. Project Management Institute - PMI, A Guide to Project Management Knowledge Body, The PMBOK Guide, 2017.
23. K. Schwalbe, "An Introduction to Project Management: With Brief Guides to Microsoft Project 2013", CreateSpace Independent Publishing Platform, p. 524, 2013.
24. K. Bhavsar, V. Shah and S. Gopalan, "Scrum: An Agile Process Reengineering in Software Engineering", International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 9, no. 3, 2019, ISSN: 2278-3075.
25. J. P. Lewis, "A Hands-on Guide to Bringing Projects in On Time and On Budget", Tata McGraw-Hill Education Private Limited, New Delhi, p. 592, 2010.
26. D. B. Hanchate, D. M. Padulkar and A. S. Shinde, "Impact of risk factors in risk management by Bayesian learning", ACM Proceedings of International Conference on Advances in Computing, ICAC-2008, p. 1–3, 2008.
27. W. Wei, and M. E. Rana, "Software Project Schedule Management using Machine Learning & Data Mining", International Journal of Scientific & Technology Research, vol. 8, no. 9, pp. 1385-1389, 2019, ISSN 227-8616.
28. A. Saranya and A. Sumithra, "Analyse the Convergence of Multi-Domain Research Opportunities in Software Engineering", International Journal of Engineering Research & Technology (IJERT), vol. 08, no. 8, 2019.
29. N. Nascimento, C. Lucena, P. Alencar and D. Cowan, "Software Engineers vs. Machine Learning Algorithms: An Empirical Study Assessing Performance and Reuse Tasks", IEEE Transactions on Software Engineering, 2018 (arXiv Organization)
30. O. Vandecruys, D. Martens, B. Baesens, C. Mues, M. D. Backer and R. Haesen, "Mining Software Repositories for Comprehensible Software Fault Prediction Models", Journal of Systems and Software, vol. 81, no. 5, pp. 823–839, 2008.
31. K. Bhavsar, V. Shah and S. Gopalan, "Scrum Challenges: An Agile Process Reengineering in Software Engineering", International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 9, no. 3, 2019, ISSN: 2278-3075.
32. J. O. Kephart, "Research challenges of autonomic computing", in Software Engineering. In proceedings 27th International Conference on Software Engineering, 2005. ICSE 2005, Saint Louis, MO, USA. IEEE, pp. 15–22, 2005, DOI: 10.1109/ICSE.2005.1553533.
33. S. A. Christy and R. Arunkumar, "Machine Learning Based Classification Models for Financial Crisis Prediction", International Journal of Recent Technology and Engineering (IJRTE), vol. 8, issue 4, pp. 4887-4893, 2019. ISSN: 2277-3878, DOI:10.35940/ijrte.D8362.118419
34. A. K. Rai, S. Agarwal, M. Khaliq, A. Kumar, "Quantitative Analysis of Development Environment Risk for Agile Software through Machine Learning", International Journal of Recent Technology and Engineering (IJRTE), vol. 7, issue 6, pp. 83-89, 2019. ISSN: 2277-3878.
35. M. Mohri, A. Rostamizadeh, A. Talwalkar, "Foundations of Machine Learning". USA, Massachusetts: MIT Press, 2012. ISBN 9780262018258.
36. E. Alba and J. F. Chicano, "Software project management with GAs", SceinceDirect, Information Sciences, vol. 177, no. 11, pp. 2380–2401, 2007.
37. M. K. Gopal and M. Amirthavalli, "Applying Machine Learning Techniques to Predict the Maintainability of Open Source Software", International Journal of Engineering and Advanced Technology (IJEAT), vol. 8, no. 5S3, pp. 195-195, 2019. ISSN: 2249–8958, DOI: 10.35940/ijeat.E1045.0785S319
38. M. Kumar and A. J. Singh, "Performance Analysis of Students Using Machine Learning & Data Mining Approach", International Journal of Engineering and Advanced Technology (IJEAT), vol. 8, no. 3, pp. 75-79, 2019. ISSN: 2249–8958.
39. R. P. Ram Kumar, S. Polepaka, S. F. Lazarus and D. V. Krishna, "An Insight on Machine Learning Algorithms and its Applications", International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 8, issue 11S2, pp. 432-436, 2019, ISSN: 2278-3075.

40. S. Bhutada and S. Morey, "Use of Machine Learning for an Automated Approach to Human Capabilities Screening", International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 8, issue 6S3, pp. 408-412, 2019, ISSN: 2278-3075.
41. D. B. Hanchate and R. Bichkar, "The machine learning in software project management: A journey. Part I", Applied Discrete Mathematics and Heuristic Algorithms, vol. 1, issue 3, pp. 21-47, 2015
42. Z. Wan, X. Xia, D. Lo and G. C. Murphy, "How does Machine Learning Change Software Development Practices?", in IEEE Transactions on Software Engineering. pp. 01-14, 2019. DOI: 10.1109/TSE.2019.2937083

## AUTHORS PROFILE

**Krunal Bhavsar** is a research scholar and corresponding author for this research paper. He is pursuing his PhD in the stream of Computer Science and Engineering from Indus University (Ahmedabad, India). Krunal has received his MBA in Information Technology from Indus University (Ahmedabad, India) in 2018 and MSc in Mobile Application and Technology (Gold Medalist) from Gujarat University (Ahmedabad, India) in 2014 and BSc in Computer Science from Gujarat University (Ahmedabad, India) in 2012. He is an enthusiastic research scientist with 14+ years of corporate experience in software business process management, software project management, software engineering and data analytics. He is certified Scrum Master by Scrum.org for certifications 'Professional Scrum Master - I' and 'Professional Scrum with Kanban - I'. He is a Technical Project Manager in software research and development organization and he has professional expertise in data science and analysis also. He has been serving as a visiting lecturer at Computer Science department at Gujarat University also. Teaching, Training, Learning are his hobbies and he is passionate about research and development in Artificial Intelligence and Machine Learning technologies.

**Dr. Vrutik Shah** is a research guide and contributor author to this research paper. He has received his PhD from Karpagam University in the stream of Computer Science and MCA in the stream of Computer Application from Nirma University. He is a Technical Project Manager in multinational well-known software development organization – Streebo and he has been leading software process redesign and redevelopment with the help of trending automation technologies. He is a certified Agile Scrum Master and having expertise in software project and product management, business analysis and business process reengineering.

**Dr. Samir Gopalan** is a research co-guide and contributor author to this research paper. He is a director and a head of department of management, external relation manager and dean of faculty (OSD Research) at Indus University. He has achieved Ph.D., M.Phil. (Computer Science), MBA (Finance) and BSc degrees in his academic career. His domain research areas are marketing management, business intelligence, training & development, skill & performance enhancement with 15+ years of experience.

[1]Krunal Bhavsar is the author to whom all the correspondence regarding this research paper should be addressed; E-mail: krunalbhavsar@engineer.com; krunalbhavsar.rs@indusuni.ac.in. Tel: +91-9737007007