Trend Application of Machine Learning in Test Case Prioritization: A Review on Techniques

Muhammad Khatibsyarbini\textsuperscript{1}, Mohd Adham Isa\textsuperscript{1}, Dayang N. A. Jawawi\textsuperscript{1}, Muhammad Luqman Mohd Shafie\textsuperscript{1}, Wan Mohd Nasir Wan Kadir\textsuperscript{1}, Haza Nuzly Abdull Hamed\textsuperscript{1}, Muhammad Dhiauddin Mohamed Suffian\textsuperscript{2}

\textsuperscript{1}School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia, 81310 Johor Bahru, Johor, Malaysia
\textsuperscript{2}Business Solution and Services, MIMOS Technology Solutions Sdn. Bhd., 57000 Kuala Lumpur, Malaysia

Corresponding author: Muhammad Khatibsyarbini (e-mail: fkmuhammad4@gmail.com).

The work is fully funded by Universiti Teknologi Malaysia (UTM) under the Industry-International Incentive Grant Scheme (IIIG) Vote Number: Q.J130000.3651.03M07.

ABSTRACT Software quality can be assured by passing the process of software testing. However, software testing process involve many phases which lead to more resources and time consumption. To reduce these downsides, one of the approaches is to adopt test case prioritization (TCP) where numerous works has indicated that TCP do improve the overall software testing performance. TCP does have several kinds of techniques which have their own strengths and weaknesses. As for this review paper, the main objective of this paper is to examine deeper on machine learning (ML) techniques based on research questions created. The research method for this paper was designed in parallel with the research questions. Consequently, 110 primary studies were selected where, 58 were journal articles, 50 were conference papers and 2 considered as others articles. For overall result, it can be said that ML techniques in TCP has trending in recent years yet some improvements are certainly welcomed. There are multiple ML techniques available, in which each technique has specified potential values, advantages, and limitation. It is notable that ML techniques has been considerably discussed in TCP approach for software testing.

INDEX TERMS Machine Learning, Software Engineering, Software Testing, Systematic Literature Review; Test Case Prioritization

I. INTRODUCTION Software engineering is not just about programming and software development. Software engineering itself is an implementation of engineering procedures in the development of any software in a systematic way \cite{1}. Within the software development process, software testing consumes a long time for execution and can be the most expensive phase \cite{2}. Software testing itself is normally carried out repetitively, even under time constraints and fixed resources. Software engineering groups are regularly compelled to end their testing activities because of financial and time requirements, which causes difficulties such as problems with software quality and client agreements.

Regression testing is an activity that confirms that new versions do not harm the previously functioning software \cite{3}, \cite{4}. As the software evolves, the software test suite has the tendency to increase in size, which frequently makes it expensive to execute. Research shows that regression testing is an expensive process that may require more than 33\% of the cumulative expenses of the software \cite{5}. In the work of Yoo and Harman \cite{6}, various regression test approaches were examined to supplement the importance of the accumulated test suite in regression testing. Those studies were then classified into three domains: minimization, selection, and prioritization. Test case prioritization (TCP) aims to order a set of test cases to achieve early optimization based on preferred properties \cite{1}, \cite{7}. It gives an approach the ability to execute first test cases that are highly significant according to some measure, and produce the desired outcome, such as revealing faults earlier and providing feedback to the testers. TCP also helps to find the ideal permutation of a series of test cases and can be executed accordingly \cite{6}.

Artificial intelligence (AI) techniques have been successfully used to reduce the effort required to carry out many software engineering activities \cite{8}. In particular, ML techniques, which belong to a research field at the intersection
of AI, computer science, and statistics, have been applied to automate various software engineering activities [9]. In a TCP approach, ML techniques have been welcomed in recent years [9]–[11]. As software systems become more complex, some conventional TCP approaches may not scale well [12]. This snowballing complexity has solidified the need for ML techniques in TCP. Even though there have been numerous studies on ML techniques in TCP, there are no advanced literature reviews that illustrate the importance of recent ML techniques for TCP. Therefore, this review paper attempts to show the trends application of ML techniques in TCP.

The point of an review paper is not to simply summarize all current proofs based on research questions, but also to bolster the improvement of evidence-based research recommendations for researchers [13]. This paper is structured as follows: Section 2 considers previous studies related to TCP approaches. Section 3 describes the strategy embraced to direct this review method. Next, results and discussion based on the research questions are presented in Section 4. Research findings are then elaborated in Section 5. In Section 6, the validity threats of this paper are discussed. Finally, Section 7 presents conclusions for this review.

II. BACKGROUND STUDIES

This section discusses prior studies to relate the review paper to the application of ML techniques in TCP. It is apparent that there have been systematic reviews that covered most TCP approach domains. However, there have been no reviews focusing specifically on ML techniques within the TCP approach itself, as ML has been trending in almost all other domains. Therefore, the authors have gathered three review studies and three mapping studies to determine the requirements of this review paper on ML techniques in TCP. A summary of nominated studies is tabulated in Table 1.

In Table 1, the first-ranked review study was done by Khatibsyarbini et al. [1], and offered a systematic review of TCP specifically for the approaches available within the domain. This study reviewed 69 studies from 1999 to 2016. Of these 69 works, more than half were taken from high-impact journals, and the rest were from either conferences or symposiums. The review resulted in several findings, and the main finding was that there were many TCP approaches. Each TCP approach specified potential values, advantages, and limitations. The review also found that the search-based TCP using ML techniques showed the most improvement in TCP regression in several recent studies.

The second review paper, authored by Arora et al. [14], covered regression testing and ML over a time period from 2000 to 2016. The majority of the studies within the work were focused on agent-based approaches in regression testing. The findings were highly related to trends and the state of the art of agent-based approaches in regression testing. The paper explored 115 studies, but only 56 studies discussed agent-based software testing, which is partially related to our review study, as this paper focuses on ML in TCP software testing. To pinpoint the finest ML technique for TCP software testing, further reviews of ML in TCP are needed, as ML techniques have been trending in various domains.

| Study type | Study reference | Study focus | Year publication | Total studies review | Years cover |
|------------|-----------------|-------------|------------------|---------------------|-------------|
| Review     | Khatibsyarbini et al. [1] | Test case prioritization | 2018 | 69 | 1999 - 2016 |
| Review     | Arora et al. [14] | Regression Testing + machine learning | 2018 | 115 | 2000 - 2016 |
| Review     | Saeed et al. [15] | Test case prioritization + machine learning | 2016 | 72 | 1975 - 2012 |
| Review     | Mece et al. [9] | Test case prioritization + machine learning | 2020 | 15 | 2006 - 2018 |
| Mapping    | Catal and Mishra [16] | Test case prioritization | 2013 | 120 | 2001 - 2011 |
| Mapping    | Durelli et al. [17] | Machine learning + software testing | 2019 | 48 | 1995 - 2018 |
| Mapping    | Prado et al. [18] | Test case prioritization + Continuous Integration | 2020 | 35 | 2009 - 2019 |

The next review paper was done by Saeed et al. [15], and deals with ML and software testing. Again, as with previous papers, this work was done in 2016 covering a time span from 1975 to 2012. This work has review 72 primary studies which mainly discuss ML in software testing. The work objectively studies the current state of the art of empirical experimentation with search-based techniques that focus on model-based testing. The results indicate that there were many works that applies AI techniques in model-based testing to achieve functional and structural coverage. The paper also concluded that there was a need for an extensive systematic analysis of the taxonomy of search-based techniques to reveal the limitations and advantages of AI application. As for the last review paper by Mece et al. [9], the paper discuss on TCP with application of ML. This work only reviews 15 primary studies cover from 2006 until 2018. The outcome of this paper manages to give a glimpse of some of ML application in TCP.

In addition to these three review studies, three mapping studies were selected for authors to better articulate relevant
A good review paper study requires a clean research method to search for and examine required prior works. With specific goals in mind, a design method as shown in Fig. 1 was systematically carried out to complete this review study. This method was inspired by Khatibsyarbini et al. [1] and Kitchenham [19].

Referring to Fig. 1, there are four main phases within the review protocol, itemized as follows: research questions, search strategy, study selection, and data synthesis and extraction. In the first phase, the research questions to be designed were based on the findings that were uncovered from the prior studies. This phase involves identifying and framing research questions based on the initial overview of the prior work. The research questions guide the selection of studies and the data extraction process.

### III. RESEARCH METHOD

A good review paper study requires a clean research method to search for and examine required prior works. With specific goals in mind, a design method as shown in Fig. 1 was systematically carried out to complete this review study. This method was inspired by Khatibsyarbini et al. [1] and Kitchenham [19].

Referring to Fig. 1, there are four main phases within the review protocol, itemized as follows: research questions, search strategy, study selection, and data synthesis and extraction. In the first phase, the research questions to be designed were based on the findings that were uncovered from the prior studies. This phase involves identifying and framing research questions based on the initial overview of the prior work. The research questions guide the selection of studies and the data extraction process.
the prior works discussed in Section 2. Seven main research questions were created to answer the uncovered findings. After the research questions were stated, a search strategy was employed that comprised specific search strings and search processes. The output of the search stage was then moved to the study selection phase. In this phase, the outcome of the search process was subject to inclusion and exclusion criteria to extract relevant studies. Quality assessments were then carried out to further evaluate the scrutinized studies. Finally, the last phase dealt with data synthesis and the extraction of primary studies that were utilized for this review study.

Referring to Figure 1, there are four main phases within the review protocol, itemized as follows: research questions, search strategy, study selection, and data synthesis and extraction. In the first phase, the research questions to be designed were based on the findings that were uncovered from the prior works discussed in Section 2. Seven main research questions were created to answer the uncovered findings. After the research questions were stated, a search strategy was employed that comprised specific search strings and search processes. The output of the search stage was then moved to the study selection phase. In this phase, the outcome of the search process was subject to inclusion and exclusion criteria to extract relevant studies. Quality assessments were then carried out to further evaluate the scrutinized studies. Finally, the last phase dealt with data synthesis and the extraction of primary studies that were utilized for this review study. The detail review protocol process was carried out by whom and how much time was cost is tabulated in Table A4 in Appendix section.

A. RESEARCH QUESTIONS STAGE
This review study aims to grasp and analyze recent experimental evidence regarding ML technique in TCP regression testing with respect to the most recent technique for further investigation as the end goal is to improvise the ability of present technique. Simultaneously, the authors wish to review the empirical evaluations used in each reviewed approach. To accomplish this goal, four main research questions with respective motivations were articulated as presented in Table III.

| Table III | RESEARCH QUESTIONS AND MOTIVATIONS |
|-----------|------------------------------------|
| **RQs**   | **RQ Statement**                     | **Motivation**                      |
| RQ 1      | What are the taxonomies ML techniques in TCP? |
| RQ 1.1    | What is the research trend of ML techniques in TCP? |
| RQ 1.2    | What is the distribution of ML techniques in TCP and its reasoning? |
| RQ 2      | What are the differences in terms of approaches for each ML technique in TCP? |
| RQ 2.1    | What are the metaphors, strength, and restrictions of existing ML techniques? |
| RQ 2.2    | How were ML technique applied and how did they affect TCP results? |
| RQ 3      | What are the processes involved in ML technique in TCP? |
| RQ 4      | What is the state of art evaluation method used for ML techniques in TCP? |
| RQ 4.1    | What and which subject study used respectively to ML techniques in TCP? |
| RQ 4.2    | What evaluation metrics used in ML techniques in TCP? |

FIGURE 1. Phases of review protocol.
All these research questions are relatively associated and concurrently explored in order to frame the objective of this review study. The uncovered and extra findings from Table II that covered by this paper will be answered by these research questions from Table II. To make things clearer, Table IV show the mapping of the uncovered and extra findings to its corresponding research questions.

As for Table IV, each research question manages to answer uncovered findings from previous works. The question was designed based on the uncovered findings also manages to provide some extra findings which serve as added value to this review study. In short, the research questions do have significance values which might be useful for other future works in ML technique in TCP related domain.

Generally, the value of review paper is realized according to the primary studies nominated. The main strategy is to have a good search string and process. In order to make searching process successful, the first thing required is the search string to be used. Not having a good search string may lead to irrelevant outcome. Therefore, the search string formulated in this study followed systemic method which consist of the following criteria:

a) Terms related to machine learning in TCP approach.
b) Terms related to specific research questions.
c) Terms with equivalent words.
d) Usage of the Boolean ‘OR’ and ‘AND’ operators as link between terms.

Since the main focus this paper to examine ML technique in TCP area, some of the results from previous studies were utilized to handpicked significant studies. “Machine learning” and “test case prioritization” are among the exact phrase utilized by authors in the most of the search queries made. The other aspect of string formulated, the search strings were made directly connected to the respective research questions. Table V show the connected search string with its respective research questions.

### TABLE IV

| Research Questions | Uncovered Findings Answered | Extra Findings | Significance of The Findings |
|--------------------|-------------------------------|----------------|-------------------------------|
| RQ1                | Trends and its reasons of ML techniques in TCP approaches. | Distribution of ML techniques in TCP approach and its verdict. | Detailed taxonomies of ML techniques in TCP approaches with its justification. |
| RQ2                | A detailed overview ML technique and empirical evidence for the techniques. | Strength and limitation of ML techniques in TCP approaches. | Provide a glimpse on how each ML technique works, and aid the research essential information for any improvement. |
| RQ3                | Specific ML technique process in TCP approach. | The differences between two process, supervised and unsupervised ML techniques in TCP. | To demonstrate the basic process of ML technique in TCP execution for ease other works make adjustment. |
| RQ4                | Dataset and evaluation metric used preferred for ML techniques in TCP approach. | The category of evaluation method used in ML techniques in TCP approach. | Information of available evaluation method which comprising study program type and scale and the evaluation metric category preferred. |

### TABLE V

| Research Questions | Related Terms | Search Strings |
|--------------------|---------------|----------------|
| RQ1                | Machine learning technique | “Machine learning technique” AND “test case prioritization” AND “Machine learning category” AND “test case prioritization” With exact phrase anywhere in the article “Classification” AND “test case prioritization” AND “advantages” “Clustering” AND “test case prioritization” AND “advantages” “reinforcement learning” AND “test case prioritization” AND “advantages” “Classification” AND “test case prioritization” AND “limitations” “Clustering” AND “test case prioritization” AND “limitations” “reinforcement learning” AND “test case prioritization” AND “limitations” |
| RQ2                | Classification technique Clustering technique Reinforcement learning Advantages or strength Limitation or weakness Test case prioritization | “Classification” AND “test case prioritization” AND “process flow” “Clustering” AND “test case prioritization” AND “process flow” |
| RQ3                | Classification technique Clustering technique Process flow Test case prioritization | “Dataset” AND “test case prioritization” AND “evaluation metric” “Case study” AND “test case prioritization” AND “evaluation metric” |
| RQ4                | Test case prioritization Evaluation metric Study program Dataset Case study | “Dataset” AND “test case prioritization” AND “evaluation metric” “Case study” AND “test case prioritization” AND “evaluation metric” |

### B. SEARCH STRATEGY STAGE

A review study required a decent search strategy as it is the key to ensure the broadness of the nominated studies.

---

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/
Prospective papers underwent a selection stage. This either the study meet the terms related to the research questions. Each research question does have several related terms used. It is also noticeable that authors utilize an exact phrase “test case prioritization” in all search string combined with other related terms. This is due to avoid the search engine return unnecessary and unrelated result with TCP domain.

C. STUDY SELECTION STAGE

As mentioned previously, to have a high impact review paper it is required to be conducted in an appropriate manner. Therefore, to make the primary studies selection, all the prospective papers gathered underwent a selection stage. This selection stage comprises with two selection phase which name inclusion and exclusion criteria and quality assessment. The process of this stage is depicted in Figure 2.

**FIGURE 2. Study selection process stage.**

From Figure 2, the process of selection of primary study start with the prospective papers gathered go through inclusion and exclusion criteria phase. The output from the phase were then scrutinize again using quality assessment where then lead toward primary study selection. The inclusion and exclusion criteria used in this review study were tabulated in Table VI, while for the quality assessment tabulated in Table VII.

**TABLE VI INCLUSION AND EXCLUSION CRITERIA**

| Inclusion Criteria                                   | Exclusion Criteria                                      |
|------------------------------------------------------|---------------------------------------------------------|
| Publication must be in English language             | Non-English language publications                      |
| Focusing on machine learning technique in test case prioritization. | Focus out from test case prioritization approach.     |
| Paper with complete bibliography information        | Paper without bibliography information                 |
| Able to answer at least one research question.       | Duplicate studies (latest paper selected)              |

The inclusion and exclusion criteria were applied to see either the study meet the terms related to the research questions, while the quality assessment intended to make sure the study selected at least manage to answer two to three research question appropriately. After the inclusion and exclusion phase, quality assessment was applied. The quality assessment of the selected studies was accomplished by scrutinize the nominated studies either they are adequate enough to answer all the RQ.

**TABLE VII QUALITY ASSESSMENT QUESTIONS**

| No | Question                                                                 |
|----|--------------------------------------------------------------------------|
| 1  | Were the paper able to answer more than two research questions?          |
| 2  | Were the paper run on complete experiment?                              |
| 3  | Does the publication publish in appropriate manner?                      |
| 4  | Were the publication have significant contribution?                      |

Authors have tabulated four quality assessment questions shown in Table VII in order to evaluate the nominated papers. The results of quality assessment were tabulated in Table A1 in Appendix section. Subsequently, some papers were rejected from this assessment phase. Upon the completion of this selection stage, 110 studies were recognized to manifest the capability to answer all of the research questions derived earlier. The inclusion and exclusion criteria were applied to see either the study meet the terms related to the research questions, while the quality assessment intended to make sure the study selected at least manage to answer two to three research question appropriately.

D. DATA SYNTHESIS AND EXTRACTION STAGE

The final stage of this research method is the data synthesis and extraction stage. The synthesis and extraction method were made correspondingly with the derived research questions. This strategy actually already applied in search string and search process where the searching process has been made with specific aim for specific data type required for each research question. Consequently, this process does benefit data extraction phase to answer each research questions. The data collected for each research question were tabulated in Table VIII.

**TABLE VIII DATA COLLECTION FOR EACH RESEARCH QUESTIONS FRAMED**

| Research Questions | Type of data extracted                                      |
|--------------------|-------------------------------------------------------------|
| RQ1                | Machine learning technique                                 |
|                    | Machine learning category                                   |
|                    | Bibliographic reference                                     |
| RQ2                | Advantage and limitation of classification technique        |
|                    | Advantage and limitation of clustering technique            |
|                    | Advantage and limitation of reinforcement learning          |
| RQ3                | Process flow of classification technique                    |
|                    | Process flow of clustering technique                        |
| RQ4                | Evaluation metric                                           |
|                    | Study program                                               |
|                    | Dataset                                                     |
|                    | Case study                                                  |
IV. RESULT AND DISCUSSION
This section outlines the results with respect to the research questions. The summary of the primary studies was presented first, followed by each research question, answered in different sub-section.

A. OVERVIEW OF PRIMARY STUDIES
Figure 3 show the percentages of collated studies. For the overview collated studies, 110 primary studies in total were nominated for this review. From the primary studies, there were 58 journal articles, 50 conference papers and 2 others articles. All the studies then were analysed and discussed under research question that been discussed previously. The percentage of the collated studies shown in Figure 3 while for the detail overviews of selected studies, Table A2 in Appendix section tabulated the information.

B. WHAT IS THE RESEARCH TREND OF MACHINE LEARNING IN TCP? (RQ1.1)
As search based TCP approach has been quite popular in recent years [1], [20], [21], the application of AI in TCP was then suggested to be assessed in a comprehensive context. Since AI quite big to be cover in single review study, only ML techniques taxonomy in TCP will be covered. The first RQ is to find the taxonomy of ML in TCP. As for the first aspect of first research questions was to examine the current publication trend regarding ML technique in TCP studies. The trend of paper published per year is depicted in Figure 4.

From the figure 4, the number of papers through the years shows a consistent increment begin from 2004 up until 2021. As the day progress, there were many new ML techniques were introduced. All these ML techniques can be categorized in several category [22]. Work by Durelli et al. [17], suggested that there were as many as five categories of ML. However, two out of five was supervised combination on semi-supervised category which have only one reference only. Therefore, authors agreed to have only three main categories in ML within TCP approach regression testing. The three categories named by supervised, unsupervised and reinforcement. Figure 5 shows the taxonomy of ML in TCP with its respective techniques.
learning is a goal oriented algorithms which learn how to achieve a specific goal or to help maximize the cumulative reward in an environment where software agent take actions [36]–[38]. Q-learning and neural network are among the popular algorithm within reinforcement learning [39]–[41]. In short, each of these three categories present different learning process depending on available dataset.

C. WHAT IS THE DISTRIBUTION OF ML TECHNIQUES IN TCP AND IT REASONING? (RQ1.2)

As for the second aspect of first research question, the RQ required a discussion on which ML technique were most utilized and why does it been chosen. The distribution for each technique is illustrated in Figure 6. The list of prior works selected for each discovered ML technique in TCP is tabulated in Table A3 in Appendix section.

From Figure 6, the results showed that classification machine learning technique is the most utilized among the selected studies. It takes 38% from the collated studies. As we know, classification technique lies under supervised category which within the category there were several algorithms could be used including Bayesian Network [32], [42]–[44], Swarm Intelligence [45]–[49], Fuzzy [50], [51] and others[52]. There were some observations are noted for classification technique utilization. Firstly, classification technique required training data which in TCP empirical data normally come with historic version which can serve as their training data [1], [17]. Second, classification target to predict discrete value which highly compatible with TCP aim which ideally to find which test cases faulty or not.

The second largest utilized technique reported in collated studies is clustering techniques with 32% contributed by these notable works [34], [51], [53]–[57]. Clustering technique look like classification which aim to grouping the inputs but they difference in term of the needs of training and testing dataset. Clustering lie in unsupervised category which has been identified in previous sub-section 4.2. Unsupervised clustering technique complexity is far less complex in compared to classification technique which considered to be the reason this technique been selected. Apart from that, not having a training and testing dataset could reduce time and resources for more cost effective TCP which can be noted for clustering technique utilization [53], [58].

Reinforcement learning technique comes as the third most utilized technique reported from the collated studies with 17% portion. The authors believe this technique able to hit such a number as the researchers [59]–[63] works on continuous integration which is a situation condition in TCP. A part from that a multi-objective TCP also play main role to have this techniques reinforcement learning been selected as this technique help maximize the cumulative reward in an environment where software agent take actions [36]–[38].

Regression and dimensional technique which have 6% and 7% portion correspondingly, which lose miserably to their superior technique within their respective category. Regression technique which categorized under supervised ML has only 6% utilization [24], [48], [64]–[66] as the technique dependent on numerical in compare to classification which dependent on categorical. Regression technique is more on statistical analysis in order to reveal the relationship between independent variables and dependent variables [67]. As for dimensional reduction, having only 7% portion did not seem to be much known but still have its own fans [68]–[70]. Authors believe this may due to the availability of other technique in TCP is much more superior and easier to access. However, the gap of this distribution percentage is getting closer. Figure 7 show the modern trend of ML techniques in TCP.

From Figure 7, the line chart shows the modern trend of ML techniques in TCP. Even though classification techniques can be considered as the most utilized techniques in TCP based on Figure 6, the number of classification work in TCP has drastically decline from nine in 2019 to only two in 2020 and 2021. The decline in number of classifications techniques in recent studies can be deduced as the technique already pass its state-of-the-art phase which mean it can be consider as an established technique in TCP. Clustering techniques on the
other hand, the trend seems to be able maintain higher than the others for final two years. As for the other techniques, the trend still on sideways mode.

**D. What are the metaphors, strength, and restrictions of existing ML techniques? (RQ2.1)**

The second research question aims to see the differences of ML techniques in TCP. As for the first aspect of second research question, the metaphors for each ML techniques as illustrated in Figure 6 is tabulated in Table IX. The outlined of these techniques are essential, as it give an understanding on how each ML techniques work in TCP. As for strength and restrictions of each ML technique in TCP, the detailed is tabulated in separate table which is Table X. This knowledge hopefully could be served as an idea and motivation for potential improvement in the future.

| TABLE IX | OVERVIEWS OF ML TECHNIQUES IN TCP |
|---------|-----------------------------------|
| ML Techniques | Classification |
| Supervised | Supervised learning model uses training data to learn a link between the input and the outputs and make classification. |
| | Classification algorithm attempt to assess the mapping from input variable to produce isolated output variables [23]–[25]. |
| | The most popular benchmark algorithm is K Nearest Neighbours [26]. |
| Regression | As regression algorithms attempt to assess the mapping from input variable to produce continuous output variables [25], [28], [29]. |
| Unsupervised | Clustering |
| | Clustering algorithms attempt to group (called cluster) object while making sure each objects from different cluster are not similar [31], [32]. |
| | Defining the distance among the object is crucial part to achieve a perfect clustering process [71]. |
| | There were many clustering algorithms available in the literature, K-Means can be said as the most popular algorithm among the researchers to be taken as their benchmark [34], [35]. |
| Dimensional Reduction | Dimensionality reduction is a technique in ML that lessens the number of test cases in TCP test suites [70]. |
| | Reduced test cases mean reduced time execution lead to cost effective [72]. |
| Reinforcement Learning | Reinforcement learning is a goal oriented algorithms which learn how to achieve a specific goal or to help maximize the cumulative reward in an environment where software agent take actions [36]–[38]. |

| TABLE X | THE ADVANTAGES AND LIMITATION OF ML TECHNIQUES IN TCP |
|---------|------------------------------------------------------|
| ML Techniques | Advantages | Limitations |
| Supervised Classification | Help with complex decision-making problems [73]. | - The number of class chosen could also have affected the results. |
| | Better at finding more faults earlier in high-risk components than other techniques [5], [1]. | - High complexity lead to high resources required [76]. |
| | Major benefit on coverage and fault detection [74], [75]. |
| Regression | High coverage result [77]. | - Time consuming [42]. |
| Unsupervised Clustering | Claim to have high efficiency in term of time execution [71], [78]. | - The speed of execution slows down [74]. |
| | Encourage cost-awareness [34], [79]. |
| Dimensional Reduction | Reduced time execution as reduced test suites [72]. | - Low coverage results as the dataset were unsupervised and hard to track back the coverage [34], [80]. |
| | - The number of class chosen could also have affected the results [34]. |
| Reinforcement Learning | Help maximize the cumulative reward in an environment where software agent take actions [36]–[38]. | - Can lead to an excess of conditions and may reduce the accuracy of results [38], [59]. |
| | Useful for continuous integration TCP [59]. |
E. HOW WERE ML TECHNIQUE APPLIED AND HOW DID THEY AFFECT TCP RESULTS? (RQ2.2)

As for the second aspect for second research question, to answer this question the selected studies were examined deeper into their experimental setup and results. For each ML techniques, authors select certain work to be elaborated in order to give a glimpse on the application of the techniques and how it affects TCP results.

Supervised ML technique

Supervised ML technique is a technique which utilized history or training data to be used in later classification process [81]. As in TCP context, most of the available dataset or study program comes with previous version which can be utilized as training data for further classification technique which far preferable compare to regression. All available previous data were analysed and trained under ML algorithms which produce a hypothesis. This hypothesis then used for classification for the current version of test case which will undergo TCP process. Work by [82], proposed a technique which utilize bug history of the software order to predict defect in the system. The model designed able to estimate fault-proneness in source code which then can be used to classify test case accordingly with coverage-based TCP approach. Recent studies show that using appropriate history can significantly coverage based TCP approach [1], [82]-[85].

Unsupervised ML technique

Unsupervised ML technique is the technique reserved when there were no historic information or incomplete information regarding study program. Unsupervised ML technique may also have been chosen as it been claim for far less complex in compare to supervised ML technique [71], [76]. Clustering technique was notable as most popular unsupervised ML technique in TCP. Work by Chen [34], proposed adaptive random sequence based on clustering techniques. By using black box information their clustering techniques manage to cluster test cases as diverse as possible. As the experiment conducted further, the result shows that the technique manages to unfold fault at earlier stage with higher effectiveness. Recent studies also show that clustering technique may have high efficiency in term of time execution which lead to cost effectiveness [58], [71].

Reinforcement Learning ML technique

As for the last technique in ML which is reinforcement learning, it may seem not very popular enough in TCP, there still some notable work [18], [38], [40], [86], which apply the technique. One of the reason of this technique been chosen was the continuous integration in TCP [18], [59], [86]. Work by [40] demonstrated reinforcement learning in TCP. This technique was introduced in order to reduce and save computing resources as the integration continuous executed. The experiment was executed using three datasets and show that reward function in reinforcement learning do have cost effect in the continuous integration environment TCP. However there also has been reported to have excessive condition during learning process may lead to reduced result accuracy [38], [59], [87].

In short, each of the .ML techniques do have advantages in different situation. Table XI summarize the suitability of techniques in different occasions.

| Techniques               | Dataset                      | Process           | Results Orientation          |
|--------------------------|------------------------------|-------------------|-------------------------------|
| Supervised               | Complete set with previous version | Waterfall Spiral | Outcome-Based |
| Unsupervised             | Complete, Incomplete         | Waterfall Spiral  | Performance-Based |
| Reinforcement Learning   | Complete, Incomplete         | Agile Continuous Integration | Performance-Based |
|                          |                              |                   | Statistical-Based |

F. WHAT ARE THE PROCESSES INVOLVED IN ML TECHNIQUE IN TCP? (RQ3)

Engineering is an art of constructing something complex look more straightforward. In this case, software engineering also does extremely concern on how the process applied throughout the software development period. Therefore, authors took initiative to investigated this kind of research question. In order to have systematic complete experiment, every experiment should follow design process to make sure the solution is run at complete satisfactory. Some of the selected studies were inspected further regarding their experiment flow. As there are two most popular ML techniques in TCP, authors able to designed standard flow of both ML techniques illustrated as in Figure 8 and Figure 9.

FIGURE 8. Standard flow process for clustering technique in TCP
As shown in Figure 8, the standard flow process for clustering in TCP have five stage while in Figure 9, classification have extra four stage before classification of test cases take place. Both of the process may start with test suites generated then move to analyse the test case information. Even though no single work clearly described these two processes, we can agree that any experiment or research activity should identify an analysed their data information first. After available information analysed, the ML technique then can be applied either clustering or classification. However, for classification do have extra work before the process can be started. Works by these researchers [74], [75], [82], [83], demonstrated few steps before classification take place. The steps are known as training phases which learn from previous version of study program or any history data which the come out with specific hypothesis. This hypothesis then used to do the classification of test suites later on. As for clustering technique there is no required pre-trained data to do the clustering. The works by researchers [34], [71], [76], [78], clearly demonstrated there were no training data required where the process directly can be started after analysed current available information. Therefore, it can be consider the main reason behind the claim that clustering technique have high efficiency in term of time execution which lead to cost effectiveness [58], [71]. After the clustering and classification test case executed, both techniques employed similar steps toward the end of the process. The next step is prioritizing the clustered or classed test case followed by evaluation of prioritized test cases.

G. WHAT AND WHICH SUBJECT STUDY TYPE USED RESPECTIVELY TO ML TECHNIQUES IN TCP? (RQ4.1)

As for the final research question which aims to unveiled the popular type of subject study utilized. There were three type subject study that normally used in any experiment or research study which can named as open-source programs, lab programs and industrial programs. The percentage of utilized study programs among selected study has been depicted in Figure 10.

From Figure 10, we can see the most used programs were open-source programs with 47% portion followed by lab program, 31% portion and industrial programs with 22% portion. Some of the open-source programs can be referred in the work of Khatibsyarbini [1]. Authors purposely to only discuss programs type used instead of listing out every programs used since most of them have been listed out and discuss in previous works [1], [14], [16], [17]. The open-source program leads the most utilized study programs as the open-source program mostly come numerous versions with various size of programs [34], [88]. As for industrial programs, authors believe the availability of industrial programs were limited for some institution which have connection directly with the industrial organization. Works by [23], [35], [61], [78] demonstrated an industrial program evaluation method where most part of the information within the programs cannot be access as confidential issues. As for lab programs, some institution may have established lab with a good team could proceed with the own study program. Also similar with the issues in industrial programs, the confidential information of the programs may reduce the availability of program to be utilized in other works [14], [57], [89], [90]. As the distribution of size of study programs used, the information illustrated in Figure 11.
From the Figure 11, open-source programs have the most number of studies in all size of study programs which have been noted as the main reasons for the most utilized study programs type in ML technique in TCP. Apart from that, Figure 11 revealed that ML technique in TCP preferred to use medium to large size of program instead of small as one of the purposes of ML itself to improve performance in term of efficiency in large scale environment. However, small scale program still reliable either to prove the concept of the ML before moving toward bigger scale of study programs.

H. WHAT KIND OF EVALUATION METRICS USED IN ML TECHNIQUES IN TCP? (RQ4.2)

In any empirical study, the most important element where could highlight either the study success or not was the results which can be determined by using several evaluation metrics. There were numerous evaluation metrics used in TCP approach. Figure 12 shows the hierarchy of evaluation method in ML technique in TCP.

From Figure 12, there were three main evaluation type which can categorize by name, statistical evaluation, performance evaluation and outcome evaluation. The main evaluation type is outcome where the evaluation was made accordingly to its main objective. Within outcome type evaluation, there were average percentage fault detected (APFD) and coverage evaluation metric which can be consider popular among the researcher in TCP domain [1], [6], [73].

Work by [1], their findings show that average percentage fault detected (APFD) was the most utilized evaluation metric across the TCP approach. APFD is a metric used to quantify how rapid a prioritized test suite detects faults which could be consider as compulsory evaluation metric in TCP [91], [92]. The values of APFD result were ranged from 0 to 1 where higher value means better faults detection rates. The equation for calculating the APFD value is shown as below.

$$APFD = 1 - \frac{TF_1 + TF_2 + \ldots + TF_m}{n \times m} + \frac{1}{2n}$$

Where $T$ is a test suite containing $n$ test cases, $F$ is a fault from set of $m$ faults revealed by $T$. $TF_1$ is the first test case in ordering of $T$ which reveals fault number $i$ and the APFD value calculated using the equation.

After outcome evaluation, empirical experiment using ML technique in TCP domain typically will highlight the performance of their techniques [39], [56], [93], [94]. This performance could be determined by the time execution of the algorithm and also by the cost involved. Whilst the evaluation stage of the experiment could stop at performance evaluation, there were few works continue with statistical evaluation. Statistical evaluation were mainly used to verify the validity of the outcome of the experiment [59], [95]. At the end it is within the choices of the researcher either to run all type evaluation available or simply go for the outcome evaluation only. As for distribution evaluation metric used in ML techniques for TCP, the data depicted in Figure 13.

V. RESEARCH FINDINGS

In the rise of machine learning in TCP domain, it is essential the knowledge of the current state of ML technique in TCP. The detailed techniques of ML within TCP are vital in order to achieve optimize TCP results. Therefore, to highlight the impact of ML technique in TCP domain, the findings for each research questions must be emphasized more. The summary
of the finding of subsequent research questions were tabulated in Table XII.

For the first research questions most of the selected studies were used to illustrate the taxonomies of ML techniques in TCP. From the results, there were three main ML techniques category and still broadly open for perfection. The publication trend of ML technique in TCP show significant improvement through the years. New ML technique using various kind of algorithm are introduced consistently almost every month. The result also show that classification technique category was the most popular follow by clustering then reinforcement learning come as the last preferred. Even though so, each of these techniques have their own supporter where does not really concern about the popularity of the technique. This can be proven by some recent publication where successfully employ reinforcement learning technique [73], [96], [97] even there were less literature available regarding the strength of the technique.

| RQs | RQ Statement | Summary Findings |
|-----|--------------|------------------|
| RQ 1 | What are the taxonomies ML techniques in TCP? | The results on trends of ML technique in TCP shows that classification come first place followed by clustering then on reinforcement learning. |
| RQ 1.1 | What is the research trend of ML techniques in TCP? | This distribution was affected by the type dataset and the numbers of literature available regarding the technique. |
| RQ 1.2 | What is the distribution of ML techniques in TCP and it reasoning? | The results on this RQ illustrated the overview of the idea on how each ML techniques works. |
| RQ 2 | What are the differences in terms of approaches for ML techniques in TCP? | Apart from that, the strength and limitation for each technique were discussed which can help other future work to select which technique suitable with their available resources. |
| RQ 2.1 | What are the metaphors, strength, and restrictions of existing ML techniques? | |
| RQ 2.2 | How were ML technique applied and how did they affect TCP results? | |
| RQ 3 | What are the processes involved in ML technique in TCP? | This special RQ which does not have any sub aspect, reveal the differences of process involved in ML technique TCP. |
| RQ 4 | What is the state of art evaluation method used for ML techniques in TCP? | The last RQ conclude that the subject study available do plays important role for the ML technique to be chosen in the first place. |

For the next research question, which intended to reveal the differences among the main available ML techniques, conclude that there were noteworthy differences in the idea of execution of ML techniques. The most notable difference was the main objective of the selected ML technique. Coverage based objective, classification technique would benefit the most [98]-[100]. As for performance wise objective, clustering technique would do the best [101]-[103]. Apart from that, the strength and limitation for each technique were discussed which can help other future work to select which technique suitable with their available resources. In short, each technique has specified potential values, benefits, and drawback.

As for the special research question which does not have any sub aspect, several studies were investigated deep into their experimental setup to give a glimpse on standard process flow in ML technique in TCP. The employment of standard process is highly essential in order to have clean project execution. The results of this research question shows that the supervised ML technique involved in training data process while the unsupervised is more straight forward. This variation of the process does profit any project manager or researcher to select which technique suite with their available resources and project schedule.

For the last research question, the results conclude that the subject study available do plays important role for the ML technique to be chosen in the first place. Medium size to large scale open source study program was consider as the most preferred due to the availability and accessibility of the study program. However, industrial study program would do better in proving the effectiveness the ML technique in real world application [47], [104]. As for the of evaluation metric, most of the previous reviews works already revealed that APFD...
was the main evaluation metric in TCP domain [1], [6], [9], [15], [16]. However, in this review study, the last research question categorizes the evaluation metric available in TCP domain specifically in ML technique into three categories. From the three categories, outcome evaluation type using APFD metric which consider the primary evaluation metric in TCP domain itself. As for ML technique works which performance wise objective would proceed with performance evaluation metric and may go for statistical evaluation to verify the results.

VI. THREAT OF VALIDITY
As a human, authors could not possibly produce a perfect review study in all aspect. Therefore, the weakness of this review study which could threaten its validity is recognized. The flaw in selecting primary studies and uncovered related field are the potential threats determined associated with human error.

A. SELECTION OF PRIMARY STUDIES
The selection of primary studies for this review paper were made with consideration in answering the designed research question respectively. In Section III, the research method used in this review study is presented in detail illustrate the process of selection of primary studies. However, in the process of the selection primary studies, it is hard for the authors to ensure all accessible works related to TCP and ML technique were reviewed. The most considerably issue can be highlight here is the numbers of research work enormously available with misleading keywords and research summary which could resulted in time wasting read through the whole research work one by one. Therefore, to encounter this issue, authors agreed to make the selection of primary study depend on specific search string connected to research question respectively.

B. UNCOVER RELATED FIELD
Within the TCP approach testing, there are several notable techniques available. However, this review study only focus on ML technique in TCP approach as ML technique which has been trending in almost other domain in recent year. Therefore, authors take initiative to investigate the state of art of ML in TCP approach to encourage the development of ML technique. In reviewing the ML technique, there were some related field not included in this review paper. The most notable uncover related field was the list of algorithms used in this ML technique. The issue here is, most of the algorithm nowadays could be tuned into different type ML technique. To make things clearer, work by [105] using neural network algorithm in classification technique, while work by [103] tweak the neural network to work on clustering technique. Therefore, to avoid misleading information, authors agreed to not list out algorithms available for each ML technique category as the algorithm can be tweak to fit the technique intended.

VII. CONCLUSION
As this paper come to the end, the purpose of this review paper has been achieved by answering all the research questions designated. The results obtained through the review study methodology scheme which required finding, categorizing and evaluating the primary studies. All this effort intended to aid other researchers to have a glimpse of current state of ML technique in TCP subsequently lead to any sort of improvement. As the result of this review, there were several notable findings which could give a guide for future work. The discovered notable findings were:

1) There several ML techniques trending in recent year yet improvement still vastly open.
2) Classification technique in ML was the most utilized as the technique benefited from the availability of historic data which resulted in high APFD and coverage effectiveness.
3) Reinforcement learning technique application required more structured process and improvement to be able to apply in standard study program.
4) Learning process time frame for ML technique could be detailed out to aid researcher or project manager making necessary tuning.

As for research suggestions, there are a few authors could suggest for future improvement in TCP. The suggested future works were:

1) A supervised and unsupervised technique that support agile or continuous changes development environment should be most welcomed.
2) A clear definition of study program size scale should be examined deeper and standardize the scale to decide whether TCP is needed or not certain size of project in future.
3) Clustering technique in ML do have performance wise and cost-effective in compared to others but still required some improvement for objective outcome results.

APPENDIX
There are three table presented here. First Table A1 shows the quality scores results of collated studies. Second Table A2 discussed on overview of collated studies. Table A3 shows total number of machine learning technique source citations. The last Table A4 show the review protocol process.
TABLE A2
OVERVIEW OF COLLATED STUDIES

| Paper Citation       | Paper Type  | Publication Year | Domain / Category |
|----------------------|-------------|------------------|-------------------|
| Chaudhary et al. [10] | Conference  | 2020             | Clustering        |
| Prado Lima et al. [11]| Journal     | 2020             | Reinforcement     |
|                      |             |                  | Learning          |
| Vescan et al. [12]   | Conference  | 2020             | Clustering        |
| Arora & Bhatia [14]  | Conference  | 2018             | Reinforcement     |
|                      |             |                  | Learning          |
| Bajaj & Sangwan [20] | Conference  | 2018             | Reinforcement     |
|                      |             |                  | Learning          |
| Ashraf et al. [21]   | Conference  | 2012             | Clustering        |
| X. Wang & Zeng [23]  | Workshop    | 2016             | Classification    |
| Lachmann et al. [24] | Conference  | 2017             | Regression        |
| Elbaum et al. [25]   | Journal     | 2004             | Clustering        |
| McRoberts, Næsset, &| Conference   | 2015             | Classification    |
| Gobakken [26]        |             |                  |                   |
| Mahdieh et al. [27]  | Conference  | 2013             | Clustering        |
| Xiao et al. [101]    | Conference  | 2015             | Clustering        |
| Fu et al. [102]      |             |                  |                   |
| Gökçe et al. [103]   | Conference  | 2017             | Clustering        |
| Shuai Wang et al. [104]| Journal    | 2018             | Clustering        |
| Gökçe & Eminli. [105]| Conference  | 2011             | Clustering        |
| Thomas et al. [106]  |             |                  |                   |
| Nascimento et al. [30]| Conference  | 2016             | Clustering        |
| Palma et al. [31]    | Conference  | 2014             | Clustering        |
| Zhao, Wang, & Wang [32]| Conference  | 2015             | Clustering        |
| Jahan et al. [33]    |             |                  |                   |
| Jinfu Chen et al. [34]| Conference  | 2018             | Clustering        |
| Carlson, Do, & Denton[35]| Journal   | 2011             | Clustering        |
| Abbeel & Ng [38]     | Conference  | 2004             | Reinforcement     |
|                      |             |                  | Learning          |
| Authors                        | Journal Year | Title                        | Source Citation | Total Citation |
|-------------------------------|---------------|------------------------------|-----------------|---------------|
| Nguyen, Le, & Nguyen          | Conference 2019 | Reinforcement Learning      | [39]            | 42            |
| Wu, Yang, Li, & Zhao         | Conference 2019 | Reinforcement Learning      | [40]            |               |
| Lousada et al. [41]          | Journal 2020  | Reinforcement Learning      |                 |               |
| Mirahbeyarbin et al. [42]    | Conference 2007 | Clustering                 |                 |               |
| Mirahbeyarbin et al. [43]    | Conference 2008 | Clustering                 |                 |               |
| Do et al. [44]               | Journal 2010  | Classification             |                 |               |
| Anki & Sehgal [45]           | Journal 2018  | Classification             |                 |               |
| Panwar, Tomar, & Singh [46]  | Journal 2018  | Classification             |                 |               |
| Bhan, Li, Zhao, & Gong [47]  | Journal 2017  | Classification             |                 |               |
| Wong, Zeng, Miao, Gao, & Yang [48] | Conference 2019 | Classification |               |               |
| Vescan et al. [49]           | Journal 2016  | Classification             |                 |               |
| Schwart & Do [50]            | Journal 2016  | Classification             |                 |               |
| Hettiarachchi et al. [51]    |                 |                             |                 |               |
| Hemmati et al. [52]          | Conference 2010 | Clustering                 |                 |               |
| Zhang et al. [53]            | Journal 2020  | Classification             |                 |               |
| Alsuakhin et al. [54]        | Conference 2017 | Regression                |                 |               |
| Miranda et al. [55]          | Conference 2018 | Classification             |                 |               |
| Khatribseyarbin et al. [56]  | Journal 2017  | Classification             | [70]            |               |
| Chaurasia & Agarwal [57]     | Conference 2016 | Clustering                 | [71]            |               |
| Harikarthik et al. [58]      | Journal 2012  | Classification             | [72]            |               |
| Jiang et al. [59]            | Journal 2019  | Classification             |                 |               |
| Qusef et al. [60]            | Journal 2014  | Clustering                 |                 |               |
| Basuagjer & Xie [61]         | Symposium 2016 |                  |                 |               |
| Thakur & Sharma [62]         | Conference 2019 | Clustering                 |                 |               |
| Junjie Chen et al. [64]      | Conference 2016 | Regression                |                 |               |
| Rosenbauer et al. [63]       | Journal 2020  | Clustering                 |                 |               |
| Singh et al. [65]            | Journal 2018  | Classification             |                 |               |
| Tonella et al. [66]          | Conference 2006 | Classification             |                 |               |
| Liu et al. [67]              | Conference 2019 | Regression                |                 |               |
| Bhargavi & Bhaskara          | Journal 2018  | Dimensional Reduction      |                 |               |
| Reddy [68]                   | Conference 2019 | Dimensional Reduction    |                 |               |
| Tahvili et al. [69]          | Journal 2018  | Dimensional Reduction      |                 |               |
| Nurmurodov et al. [70]       | Conference 2019 | Dimensional Reduction    |                 |               |
| Khalid & Qamar [71]          | Conference 2015 | Dimensional Reduction    |                 |               |
| Nagar et al. [72]            | Conference 2015 | Dimensional Reduction    |                 |               |
| Bajaj & Sangwan [73]         | Journal 2019  | Classification             |                 |               |
| Hajri et al. [76]            | Journal 2016  | Classification             |                 |               |
| D.K. Yadav & Dutta [77]      | Conference 2016 | Classification             |                 |               |
| Srikanth et al. [78]         | Journal 2016  | Classification             |                 |               |
| Lachmann [79]                | Conference 2015 | Classification             |                 |               |
| Luo et al. [80]              | Conference 2015 | Classification             |                 |               |
| Jordan & Mitchell [81]       | Conference 2015 | Classification             |                 |               |
| Mahdieh et al. [82]          | Journal 2019  | Classification             |                 |               |
| Palma et al. [83]            | Conference 2018 | Classification             |                 |               |
| Noguchi et al. [84]          | Conference 2015 | Classification             |                 |               |
| Lin et al. [85]              | Conference 2013 | Classification             |                 |               |
| Xiao et al. [86]             | Journal 2018  | Classification             |                 |               |
| Pradhan et al. [87]          | Conference 2019 | Classification             |                 |               |
| Srivastava et al. [89]       | Conference 2009 | Clustering                 |                 |               |
| Dharmveer et al. [90]        | Conference 2019 | Clustering                 |                 |               |
| Khatibseyarbin et al. [93]   | Journal 2019  | Clustering                 |                 |               |
| Luo et al. [94]              | Conference 2018 | Classification             |                 |               |
| Ledru et al. [95]            | Journal 2012  | Classification             |                 |               |
| Speiker et al. [96]          | Journal 2017  | Classification             |                 |               |
| Ponaraserti et al. [97]      | Journal 2008  | Classification             |                 |               |
| Kansaard & Ramingwong [98]   | Conference 2015 | Clustering                 |                 |               |
| Dharmveer et al. [99]        | Conference 2017 | Clustering                 |                 |               |
| Hasnain et al. [100]         | Journal 2019  | Classification             |                 |               |
| Xiao et al. [101]            | Conference 2016 | Classification             |                 |               |
| Fu et al. [102]              | Journal 2017  | Classification             |                 |               |
| Gökke et al. [103]           | Journal 2015  | Classification             |                 |               |
| Shuai Wang et al. [104]      | Conference 2016 | Classification             |                 |               |
| Gokce & Eminli [105]         | Journal 2014  | Classification             |                 |               |
| Thomas et al. [106]          | Journal 2014  | Classification             |                 |               |
| Emam & Miller [107]          | Journal 2015  | Classification             |                 |               |
| Sujata & Purohit [108]       | Conference 2017 | Classification             |                 |               |
| Harikarthik et al. [109]     | Journal 2018  | Classification             |                 |               |
| Song Wang et al. [110]       | Symposium 2017 | Classification             |                 |               |
| Eminli et al. [111]          | Conference 2006 | Clustering                 |                 |               |
| Eghbali & Tahvildari [112]   | Journal 2016  | Classification             |                 |               |
| Bania [113]                  | Conference 2019 | Classification             |                 |               |
| Kalyani et al. [114]         | Journal 2018  | Classification             |                 |               |
| Zhang et al. [115]           | Conference 2019 | Classification             |                 |               |
| Panwar et al. [116]          | Conference 2018 | Classification             |                 |               |
| Chi et al. [117]             | Journal 2020  | Classification             |                 |               |
| Mukherjee & Patnaik [118]    | Journal 2018  | Classification             |                 |               |
| S. Kumar & Ranjan [119]      | Journal 2017  | Classification             |                 |               |
| Anderson et al. [120]        | Conference 2019 | Classification             |                 |               |
| Hemmati et al. [121]         | Journal 2013  | Classification             |                 |               |
| Mojtaba et al. [122]         | Journal 2021  | Classification             |                 |               |
| Lukas Rosenbauer et al. [123]| Conference 2021 | Classification             |                 |               |
| Claudio Magalhães et al. [124]| Conference 2021 | Classification             |                 |               |
| Maral Azizi et al. [125]     | Conference 2021 | Classification             |                 |               |
| K. Hema Shankari et al. [126]| Conference 2021 | Classification             |                 |               |
| N.Gokilavani et al. [127]    | Journal 2021  | Dimensional Reduction      |                 |               |
| Stefan Mohacsi et al. [128]  | Conference 2021 | Dimensional Reduction    |                 |               |
| Song Wang et al. [129]       | Conference 2021 | Dimensional Reduction    |                 |               |
| N.Gokilavani et al. [130]    | Journal 2021  | Classification             |                 |               |
| Anu Bajaj et al. [131]       | Conference 2021 | Classification             |                 |               |
| HanyaPei et al. [132]        | Journal 2021  | Classification             |                 |               |
| Weibo Wang et al. [133]      | Conference 2021 | Classification             |                 |               |

**TABLE A3**

| Machine Learning Technique Source Citation | Total Citation |
|--------------------------------------------|---------------|
| 1 Classification                           | 42            |

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3155508, IEEE Access
### Table A4: Review Protocol Process

| Review Process                  | Author Involved                                      | Time Consumed (Estimated Days) |
|--------------------------------|------------------------------------------------------|-------------------------------|
| Design research questions       | - Muhammad Khartibyarsibini                          | 1-2                           |
|                                | - Mohd Adham Isa                                    |                               |
|                                | - Muhammad Luqman                                   |                               |
|                                | - Mohd Shafie                                       |                               |
| Review research questions       | - Dayang N. A. Jawawi                                | 1-2                           |
|                                | - Mohd Adham Isa                                    |                               |
|                                | - Wan Mohd Nasir Wan                                 |                               |
|                                | - Haza Nuzly Abdullah                                 | 1-2                           |
|                                | - Hamed                                             |                               |
|                                | - Muhammad Dhiuaddin                                 |                               |
|                                | - Mohamed Suffian                                    |                               |
| Finalize research questions     | - Dayang N. A. Jawawi                                | 1-2                           |
|                                | - Muhammad Khartibyarsibini                          |                               |
|                                | - Mohd Adham Isa                                    |                               |
| Define search strategy         | - Muhammad Khartibyarsibini                          | 1-2                           |
|                                | - Mohd Shafie                                       |                               |
|                                | - Muhammad Luqman                                   |                               |
| Define search process          | - Muhammad Khartibyarsibini                          | 1-2                           |
|                                | - Mohd Shafie                                       |                               |
|                                | - Muhammad Luqman                                   |                               |
| Primary studies searching      | - Muhammad Khartibyarsibini                          | 15-20                         |
|                                | - Muhammad Luqman                                   |                               |
|                                | - Mohd Shafie                                       |                               |
| Primary studies assessment     | - Muhammad Khartibyarsibini                          | 15-20                         |
|                                | - Muhammad Luqman                                   |                               |
|                                | - Mohd Shafie                                       |                               |
|                                | - Dhiuaddin                                          |                               |
|                                | - Suffian                                            |                               |
| Paper review and comment       | - Dayang N. A. Jawawi                                | 7-10                          |
|                                | - Mohd Adham Isa                                    |                               |
|                                | - Wan Mohd Nasir Wan                                 |                               |
|                                | - Haza Nuzly Abdullah                                 |                               |
|                                | - Hamed                                             |                               |
|                                | - Dhiuaddin                                          |                               |
|                                | - Suffian                                            |                               |

### ACKNOWLEDGMENT

The work is funded by Universiti Teknologi Malaysia (UTM) under the Industry-International Incentive Grant Scheme (IIIG) (Vote Number: Q.J130000.3651.03M07). We would also like to thank the members of Software Engineering Research Group (SERG) and Embedded & Real-Time Software Engineering Laboratory (EReTSEL), Faculty of Computing, UTM for their feedback and continuous support.

### REFERENCES

[1] M. Khartibyarsibini et al., “Test case prioritization approaches in regression testing: A systematic literature review,” *Inf. Softw. Technol.*, vol. 93, pp. 74–93, Sep. 2018. doi: 10.1016/j.infsof.2017.08.014.

[2] G. J. Myers, T. M. Thomas, C. Sandler, T. Badgrett, T. M. Thomas, and C. Sandler, *The Art of Software Testing*, vol. 1. Hoboken, NJ, USA: John Wiley & Sons, Inc., 2004.

[3] H. K. N. Leung, “Insights into Regression Testing,” *Proc. Int. Conf. Softw. Maint.*, pp. 60–69, 1989, doi: 10.1109/ICSM.1989.65194.

[4] S. Elbaum, A. G. Malishkevich, and G. Rothermel, “Test case prioritization: a family of empirical studies,” *IEEE Trans. Softw. Eng.*, vol. 28, no. 2, pp. 159–182, 2002, doi: 10.1109/32.988497.

[5] P. K. Chittimall and M. J. Harrold, “Recomputing coverage information to assist regression testing,” *IEEE Trans. Softw. Eng.*, vol. 35, no. 4, pp. 452–469, 2009, doi: 10.1109/TSE.2009.4.

[6] S. Yoo and M. Harman, “Regression testing minimization, selection and prioritization: A survey,” *Softw. Testing, Verif. Reliab.*, vol. 22, no. 2, p. n/a-n/a, 2012, doi: 10.1002/stv.430. A. Kumar and K. Singh, “A Literature Survey on test case prioritization,” 2014. Accessed: Dec. 09, 2016. [Online]. Available: http://ijact.in/index.php/ijact/article/viewFile/320/271.

[7] [8] D. Hao, L. Zhang, L. Zhang, Y. Wang, X. Wu, and T. Xie, “To Be Optimal or Not in Test-Case Prioritization,” *IEEE Trans. Softw. Eng.*, vol. 42, no. 5, pp. 490–504, 2016, doi: 10.1109/TSE.2015.2496939.

[9] E. K. Mece, K. Binjaku, and H. Paci, “The Application Of Machine Learning In Test Case Prioritization - A Review,” *Eur. J. Electr. Eng. Comput. Sci.*, vol. 4, no. 1, 2020, doi: 10.24018/ejce.2020.4.1.128.

[10] S. Chaudhary and A. Jatain, “Performance Evaluation of Clustering Techniques in Test Case Prioritization,” in 2020 International Conference on Computational Performance
[44] H. Do, S. Mirarab, L. Tahvildari, and G. Rothermel, “The Effects of Time Constants on Test Case Prioritization: A Series of Controlled Experiments,” IEEE Trans. Softw. Eng., vol. 36, no. 5, pp. 593–617, 2010, doi: 10.1109/TSE.2010.58.

[45] S. Anku and N. Sehgal, “Enhanced test case prioritization technique using bat algorithm,” 2018. Accessed: Feb. 10, 2020. [Online]. Available: www.IJARIT.com.

[46] D. Panwar, P. Tomar, and V. Singh, “Hybridization of Cuckoo-ACO algorithm for test case prioritization,” J. Stat. Manag. Syst., vol. 21, no. 4, pp. 539–546, Jul. 2018, doi: 10.1080/09720510.2018.1466962.

[47] Y. Bian, Z. Li, R. Zhao, and D. Gong, “Empirically Based ACO for Regression Test Case Prioritization,” IEEE Trans. Emerg. Top. Comput. Intell., vol. 1, no. 3, pp. 213–223, 2017, doi: 10.1016/j.tsci.2017.0699228.

[48] Y. Wong, H. Zeng, H. Miao, H. Gao, and X. Yang, “The cuckoo search and integer linear programming based approach to time-aware test case prioritization considering execution environment,” in International Conference on Collaborative Computing: Networking, Applications and Workshopping, 2019, vol. 268, pp. 734–754, doi: 10.1007/978-3-030-12981-1_51.

[49] A. Vescan, C.-M. Pintea, and P. C. Pop, “Test Case Prioritization—ANT Algorithm With Faults Severity,” Log. J. IGPL, Nov. 2020, doi: 10.1093/jigpal/zaa061.

[50] A. Schwartz and H. Do, “Cost-effective regression testing using Adaptive Test Prioritization strategies,” J. Syst. Softw., vol. 115, pp. 61–81, 2016, doi: 10.1016/j.jss.2016.01.018.

[51] C. Hettiarachchi, H. Do, and B. Choi, “Risk-based test case prioritization using a fuzzy expert system,” Inf. Softw. Technol., vol. 69, pp. 1–15, Jan. 2016, doi: 10.1016/j.infsof.2015.08.008.

[52] K. Zhang, Y. Zhang, L. Zhang, H. Gao, R. Yan, and J. Yan, “Neuron Activation Frequency Based Test Case Prioritization,” in Proceedings - 2020 International Symposium on Theoretical Aspects of Software Engineering, TASE 2020, Dec. 2020, pp. 81–88, doi: 10.1109/TASE49443.2020.00020.

[53] H. Hemmati, A. Arcuri, and L. Briand, “Reducing the cost of model-based testing through test case diversity,” in IFIP International Conference on Testing Software and Systems, 2010, pp. 63–78, doi: 10.1007/978-3-642-16573-3_6.

[54] E. Alsakhi, A. A. Saifan, and H. Alawneh, “A new data mining-based framework to test case prioritization using software defect prediction,” Int. J. Open Source Softw. Process., vol. 8, no. 1, pp. 21–41, 2017, doi: 10.4018/IOSSP.2017010102.

[55] B. Miranda, E. Cruciani, R. Verdecella, and A. Bertolino, “FAST approaches to scalable similarity-based test case prioritization,” in Proceedings - International Conference on Software Engineering, 2018, vol. 2018-Janua, 2018, pp. 1013–1022, doi: 10.1109/ICSE.2018.00028.

[56] S. Tahvili, L. Hatvani, M. Felderer, W. Afzal, and M. Bohlin, “Automated functional dependency detection between test cases using Doc2Vec and Clustering,” in Proceedings - 2019 IEEE International Conference on Artificial Intelligence Testing, AITest 2019, May 2019, pp. 19–26, doi: 10.1109/AITest.2019.00-13.

[57] B. Nurmuradov, R. Bryce, S. Piparia, and B. Bryant, “Clustering and Combinatorial Methods for Test Suite Prioritization of GUI and Web Applications,” in Advances in Intelligent Systems and Computing, 2018, vol. 738, pp. 459–466, doi: 10.1007/978-3-319-77028-4_60.

[58] Z. Khalid and U. Qamar, “Weight and Cluster Based Test case Prioritization Technique,” in 2019 IEEE 10th Information Technology, Electronics and Mobile Communication Conference, IEMCON 2019, 2019, pp. 1013–1022, doi: 10.1109/IEMCON.2019.8936202.

[59] R. Nagar, A. Kumar, G. P. Singh, and S. Kumar, “Test case selection and prioritization using cuckoo search algorithm,” in 2015 1st International Conference on Futuristic Trends in Computational Analysis and Knowledge Management, ABLAZE 2015, 2015, pp. 283–288, doi: 10.1109/ABLAZE.2015.7155012.

[60] A. Bajaj and O. P. Sangwan, “A Systematic Literature Review of Test Case Prioritization Using Genetic Algorithms,” IEEE Access, vol. 7, pp. 126355–126375, 2019, doi: 10.1109/ACCESS.2019.2938260.

[61] B. Busjæger and T. Xie, “Learning for test prioritization: An industrial case study,” in Proceedings of the ACM SIGSOFT Symposium on the Foundations of Software Engineering, Nov. 2016, pp. 975–980, doi: 10.1145/2950290.2983954.

[62] A. Thakur and G. Sharma, “Neural Network Based Test Case Prioritization in Software Engineering,” in Communications in Computer and Information Science, 2019, vol. 956, pp. 334–345, doi: 10.1007/978-981-13-3143-5_28.

[63] L. Rosenbauer, A. Stein, D. Patzel, and J. Hahner, “XCSF with Experience Replay for Automatic Test Case Prioritization,” in 2020 IEEE Symposium Series on Computational Intelligence, SSCI 2020, Dec. 2020, pp. 307–314, doi: 10.1109/SSCI47803.2020.9380379.

[64] J. Chen et al., “Test Case Prioritization for Compilers: A Text-Vector Based Approach,” in Proceedings - 2016 IEEE International Conference on Software Testing, Verification and Validation, ICST 2016, 2016, pp. 266–277, doi: 10.1109/ICST.2016.19.

[65] A. Singh, R. K. Bhatia, and A. Singhrova, “Object Oriented Coupling based Test Case Prioritization,” Int. J. Comput. Sci. Eng., vol. 6, no. 9, pp. 747–754, 2018, doi: 10.26438/jcsce/v6i9.747754.

[66] P. Tonella, P. Avesani, and A. Susi, “Using the case-based ranking methodology for test case prioritization,” in IEEE International Conference on Software Maintenance, ICSM, 2006, pp. 123–132, doi: 10.1109/ICSME.2006.74.

[67] W. Liu, J. Jiang, J. H. Yang, Y. Song, and Z. Dou, “A Baseline Evaluation Method Based on Principal Component Analysis for Software Test Case Design,” in Proceedings - Companion of the 19th IEEE International Conference on Software Quality, Reliability and Security, QRS-C2019, Jul. 2019, pp. 78–83, doi: 10.1109/QRS-C2019.00028.

[68] K. Bhargavi and T. Bhaskara Reddy, “Computationally efficient secure and privacy preserving storage ofhuge data on hybrid cloud,” J. Theor. Appl. Inf. Technol., vol. 96, no. 14, pp. 4316–4327, 2018, Accessed: Feb. 11, 2020. [Online]. Available: https://jaitj.org/volumes/ninetysix14.php.

[69] S. Tahvili, L. Hatvani, M. Felderer, W. Afzal, and M. Bohlin, “Automated functional dependency detection between test cases using Doc2Vec and Clustering,” in Proceedings - 2019 IEEE International Conference on Artificial Intelligence Testing, AITest 2019, May 2019, pp. 19–26, doi: 10.1109/AITest.2019.00-13.
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3135508, IEEE Access
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3135508, IEEE Access.

[109] S. K. Harikarthik, P. Ramanathan, and V. Palanisamy, “Enhancement of regression testing using genetic data generation and test case prioritization using m-ACO technique,” 2018. Accessed: Febr. 10, 2020. [Online]. Available: www.sciencepubco.com/index.php/IJET.

[110] S. Wang, J. Nam, and L. Tan, “QTEP: Quality-aware test case prioritization,” Proc. ACM SIGSOFT Symp. Found. Softw. Eng., vol. Part F1301, pp. 523–534, Aug. 2017, doi: 10.1145/3327310.3326258.

[111] M. Eminli, F. Belli, N. Gökoğe, and M. Eminoğlu, “Coverage-Based, Prioritized Testing Using Neural Network Clustering,” LNCS, vol. 4263, pp. 1060–1071, 2006, doi: 10.1007/11902140_110.

[112] S. Eghbali and L. Tahvildari, “Test Case Prioritization Using Lexicographical Ordering,” IEEE Trans. Softw. Eng., vol. 5589, no. January, pp. 1–1, 2016, doi: 10.1109/TSE.2016.2550441.

[113] O. Bansal, “Test case selection prioritization approach based on memoization dynamic programming algorithm,” Inf. Softw. Technol., vol. 115, pp. 119–130, Nov. 2019, doi: 10.1016/j.infsof.2019.06.001.

[114] R. Kalyani, P. Sai Mounika, R. Naveen, G. Maridu, and P. Ramya, “Test Case Prioritization Using Requirements Clustering,” 2018. Accessed: Feb. 10, 2020. [Online]. Available: http://www.ripublication.com.

[115] W. Zhang, Y. Qi, X. Zhang, B. Wei, M. Zhang, and Z. Dou, “On test case prioritization using ant colony optimization algorithm,” in Proceedings - 21st IEEE International Conference on High Performance Computing and Communications, 17th IEEE International Conference on Smart City and 5th IEEE International Conference on Data Science and Systems, HPCC/SmartCity/DSS 2019, vol. 29, pp. 2767–2773, 2019, doi: 10.1109/HPCC/SmarCity/DSS.2019.00358.

[116] D. Panwar, P. Tomar, H. Harsh, and M. H. Siddique, “Enhancement of regression testing using genetic data generation and test case prioritization using m-ACO technique,” 2018. Accessed: Feb. 10, 2020. [Online]. Available: www.sciencepubco.com/index.php/IJET.

[117] M. Eminli, F. Belli, N. Gökoğe, and M. Eminoğlu, “Coverage-Based, Prioritized Testing Using Neural Network Clustering,” LNCS, vol. 4263, pp. 1060–1071, 2006, doi: 10.1007/11902140_110.

[118] S. Eghbali and L. Tahvildari, “Test Case Prioritization Using Lexicographical Ordering,” IEEE Trans. Softw. Eng., vol. 5589, no. January, pp. 1–1, 2016, doi: 10.1109/TSE.2016.2550441.

[119] O. Bansal, “Test case selection prioritization approach based on memoization dynamic programming algorithm,” Inf. Softw. Technol., vol. 115, pp. 119–130, Nov. 2019, doi: 10.1016/j.infsof.2019.06.001.

[120] R. Kalyani, P. Sai Mounika, R. Naveen, G. Maridu, and P. Ramya, “Test Case Prioritization Using Requirements Clustering,” 2018. Accessed: Feb. 10, 2020. [Online]. Available: http://www.ripublication.com.

[121] W. Zhang, Y. Qi, X. Zhang, B. Wei, M. Zhang, and Z. Dou, “On test case prioritization using ant colony optimization algorithm,” in Proceedings - 21st IEEE International Conference on High Performance Computing and Communications, 17th IEEE International Conference on Smart City and 5th IEEE International Conference on Data Science and Systems, HPCC/SmartCity/DSS 2019, vol. 29, pp. 2767–2773, 2019, doi: 10.1109/HPCC/SmarCity/DSS.2019.00358.

[122] D. Panwar, P. Tomar, H. Harsh, and M. H. Siddique, “Enhancement of regression testing using genetic data generation and test case prioritization using m-ACO technique,” 2018. Accessed: Feb. 10, 2020. [Online]. Available: www.sciencepubco.com/index.php/IJET.

[123] L. Rosenbauer, D. Pätzel, A. Stein, and J. Hähner, “Transfer Learning for Automated Test Case Prioritization Using XCSF,” Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 12694 LNCS, pp. 681–696, Apr. 2021, doi: 10.1007/978-3-030-72699-7_43.

[124] C. Magalhães, A. Mota, and L. Momente, “UI Test case prioritization on an industrial setting: A search for the best criteria,” Softw. Qual. J. 2021, pp. 681–72699-7_43.

[125] M. Azizi, “A tag-based recommender system for regression test case prioritization,” Proc. - 2021 IEEE 14th Int. Conf. Softw. Test. Verif. Valid. Work. ICSTW 2021, pp. 146–157, Apr. 2021, doi: 10.1109/ICSTW52544.2021.00035.

[126] K. Hema Shankari, S. Mathivilasini, D. Arasu, and G. Suseendran, “Genetic Algorithm Based on Test Suite Prioritization for Software Testing in Neural Network,” Adv. Intell. Syst. Comput., vol. 1292, pp. 409–416, 2020, doi: 10.1007/978-3-033-4839-4_37.

[127] N. Gokilavani and B. Bharathi, “Multi-Objective based test case selection and prioritization for distributed cloud environment,” Microprocess. Microsyst., vol. 82, p. 103964, Apr. 2021, doi: 10.1016/J.MICPRO.2021.103964.

[128] S. Mohan and M. Felderer, “AI-Based Enhancement of Test Models in an Industrial Model-Based Testing Tool,” Proc. - 2021 IEEE Int. Conf. Softw. Anal. Evol. Reengineering, SANER 2021, pp. 636–638, Mar. 2021, doi: 10.1109/SANERS9067.2021.00080.

[129] S. Wang, N. Shreshtha, A. K. Subburaman, J. Wang, M. Wei, and N. Nagappan, “Automatic Unit Test Generation for Machine Learning Libraries: How Far Are We?,” pp. 1548–1560, May 2021, doi: 10.1109/ICSE4902.2021.00138.

[130] N. Gokilavani and B. Bharathi, “Test case prioritization to examine software for fault detection using PCA extraction and K-means clustering with ranking,” Soft Comput., vol. 25, no. 7, pp. 5163–5172, Apr. 2021, doi: 10.1007/s00500-020-05517-z.

[131] A. Bajaj and O. P. Sangwan, “Discrete and combinatorial gravitational search algorithms for test case prioritization and minimization,” Int. J. Inf. Technol., vol. 13, no. 2, pp. 817–823, Apr. 2021, doi: 10.1016/S1487-021-00628-8.

[132] H. Pei, B. Yin, M. Xie, and K. Y. Cai, “Dynamic random testing with test case clustering and distance-based parameter adjustment,” Inf. Softw. Technol., vol. 131, p. 106470, Mar. 2021, doi: 10.1016/j.infsof.2020.106470.

[133] W. Wang, Y. Wu, and Y. Liu, “A Passed Test Case Cluster Selection Method to Improve Fault Localization,” https://doi.org/10.1142/S0218126621500535, vol. 30, no. 3, Aug. 2020, doi: 10.1142/S0218126621500535.

[134] F. Yuan, Y. Bian, Z. Li, and R. Zhao, “Epistatic Genetic Algorithm for Test Case Prioritization,” Int. Symp. Search Based, 2015, Accessed: Dec. 07, 2016. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-319-22183-0_8.