



Designing a Machine Learning and Deep Learning based Quality Engineering Framework for Real-time Decision Making IT systems

Gouri Amol Vaidya

Asst. Professor, Indira College of Commerce and Science, Pune 411057, India

vaidya.gouri@iccs.ac.in

Abstract

In this paper we have done an extensive review of the work done by the researchers in the area of real-time decision making systems. The performance evaluation using ML/DL techniques have been discussed for the various quality parameters used by the researchers to monitor performance of the systems. Based on our review we concluded that very few researchers have done work on real-time decision making system and its performance analysis using DL and ML techniques. We have also stated research gaps and research questions based on our literature review. We have identified our future objectives as to find shortcomings in existing RTDM for specific IT systems, specific Machine learning and Deep learning algorithms that will support in enhancing the real-time decision making systems and to create a quality engineering architecture/framework for specific RTDM in IT systems.

Keywords: Real-time, Decision Making, non-functional requirements, Machine Learning, Deep Learning

1. INTRODUCTION

1.1 Decision making systems

Computers are being used extensively in view of organizational decision making, support and overall benefit (Turban, 2007). In group decision making, people perform group work by collaborating and communicating among one-another. Decision-making is solely an expertise obtained over a protracted duration by gaining knowledge over a vast experience. Management or industrial decision making is a kind of special artwork due to the fact that a whole lot of character patterns can be utilized for correctly fixing the identical forms of managerial problems. These patterns had been frequently primarily based totally on creativity, judgment, intuition, and enjoy instead of on systematic quantitative strategies grounded in a systematic approach (Turban, 2007). Decision making systems follows a systematic process where the experience is simulated and fed as input to the system. Such systems produce several alternatives to choose from in the real environment. While deciding the profitable outcome, some goals need to be maximized while some goals need to be minimized; as an example the costs should be maximized and risk should be minimized. In order to deal with such conflicting goals, some special multi-criteria methodologies is developed like analytic hierarchy process (AHP) (Rakesh Ranjan Kumar & Chiranjeev Kumar, 2018) where a multi criteria decision making method is proposed for cloud service ranking and selection. In the work given by (Unver, 2018), the authors implemented a problem structuring method to discover and prioritize danger elements within side the workstations of a vehicle production company. They recognized elements below 3 predominant component groups, i.e. design-driven, process-driven and human-centered elements primarily based totally at the judgments of the managers and professionals of the company. They additionally determined significance of the elements by the use of an analytic hierarchy process (AHP) method.

1.2 Performance evaluation and IT systems

While deciding the requirements of a software system, we are very much concerned about approximately translating customer requirements into functional requirements. These are the ones which describe the way system responds to the user inputs and what service it provides to the end-user; as a result deliver value to the commercial enterprise. A hit venture additionally needs to focus on non-functional requirements (NFR) that describes how effectively the software works (i.e. how much secured, scalable, maintainable, etc.). In reality NFR are an integral part of a project delivery in successful manner. They are simply extra critical because the software's functionality, making sure the software operates with the 'qualities' of the commercial enterprise needs.

Using design pattern and an object-oriented approach (Chung L., 2003) developed adaptable software architectures with a Proteus framework. The proposed method follows an NFR approach for home appliance control system (HACS). Authors have presented how to analyze and use design patterns as potential adaptability boosters in software system development. Characteristics of functional attributes addressed are communication mechanisms between devices and coordination among processes while adaptability is addressed as an NFR attributes. A usability validation check in mobile applications is studied by (Alice Mugisha, Victoria Nankabirwa, Thorkild Tylleskär, & Ankica Babic, 2019). The authors have taken case study of Mobile data capturing for health data with challenges especially in rural areas with semi-literate users. The main aim of this work was to collect the usability evaluation checklist to design Mobile Electronic Data Capturing Forms (MEDCFs) in order to improve usability. To generate the above checklist the authors have collected the views of novice as well as expert users regarding usability criteria. (Pooley & Chan, 2021) proposed a CASE study based approach to assess the success of agile methodology implementations in a gig economy by small and medium-sized enterprises(SME). The authors have investigated agile implementation in SME in short-term contracts or freelance work (gig economy), when it faces performance maintenance challenges due to resource uncertainty, to propose evaluation framework for performance assessment.

2. ISSUES ADDRESSED IN THE PAPER

2.1 Decision making and NFR

While functional requirements typically acquire extra interest at some point of the system design, there are domains wherein NFRs play a crucial position in figuring out the correctness of a system (Saadatmand M., 2015). An example of such instances is real-time applications whose correctness isn't always depending upon the correctness of the logical effects of computations, however additionally at run-time when the effects are produced. To assess the impact of NFR on designing a system it is advisable to consider multiple NFR criteria and their mutual correlation, balance and tradeoff among NFRs so as to study its combined influence on the success of the system. As the complexity of the application grows, these parameters cannot be studied individually. The author proposes a model based on fuzzy logic and DSS to aid decision making for different design alternatives for optimum values of NFR. Their work addresses the major challenges to deal with NFR which are complex relationships and inter dependencies present among them.

In the work entitled "Supporting Decision-Making for Self-Adaptive Systems: From Goal Models to Dynamic Decision Networks"; (Belaggoun, 2013) addresses the issue of uncertainty during runtime in self-adaptive systems. One of the principal ideas put forth in this paper is about making possible the best decision making with the aid of Dynamic Decision Networks (DDNs). Their technique permits the specification of choices over the NFRs and helps reasoning about limited satisfaction of NFRs by the use of probabilities. Decision-making is supported by goal models by using DDNs.

Various quality attributes of decision-making systems are analyzed by formal methods (O. N. Dolinina, 2019) to develop an expert system to detect the gas turbines faults. As a part of the experiment the problem was formulated to maximize the quality of software. The output was used to determine the plan's realization by making use of Boolean functions and expert rules.

(Gutierrez F., 2012), proposed work that addresses the challenge of designing a software system architecture which is meant to develop for partial virtual communities like community of a school/university course, of an office or even a small neighborhood. The specific examples of PVC in the current context can be ERP/Educational Management System or e-mailing system. The NFR studied for PVC supporting systems are performance, uptime (availability), maintainability and scalability, response time, extensibility, privacy, security, usability. In the work (Ruiz-López T., 2013), the author addresses Non-functional requirements of Ubiquitous systems at requirements engineering stage for Ambient Assisted Living (AAL).

(David Ameller and Claudia Ayala, 2013) have worked upon deriving and exploring the close relationship between NFR and Software Architectures. They have used NVivo software to assess the collected data. They communicate with 13 Companies of 3 different kinds like software consultancy companies, IT departments in product-based companies and software companies which design, develop particular proprietary solutions to bring them into the market. They checked how software architects elicited and documented NFRs as well as the way they tested the systems later against them. The NFRs addressed in this work are elicitation, documentation, and validation.

Architectural Decision Making (David Ameller and Claudia Ayala, 2013) is studied with respect to Non-functional Requirements by focusing on elicitation, documentation, and validation. In this paper, the author addresses the issue of uncertainty during runtime in self-adaptive systems. One of the principal ideas put forth in this paper is about making possible the best decision making by using Dynamic Decision Networks (DDNs).

2.2 Real-time systems

The work about application of ANN for prediction (Dawei Han, 2007) focuses on real-time flood forecasting using ANN techniques. This paper describes three ANN configurations-dedicated ANN, multiple output and multiple inputs and single output. The results of these three models are compared against TF model for short-range and longer-range flood forecasting. Also the training data is reused at the time of prediction to perform distance analysis at real-time prediction. Similarly response analysis is done to reveal the hydrological characteristics of ANN models

2.2.1 Real time decision making and NFR

Software architecture develops large, complex applications in order to reduce development costs with increased the quality and allows evolution. (Cooper K., Dai L., Deng Y., 2004) have worked on efficient designing of a software architecture by fulfilling its non-functional requirements at every stage of SDLC.

Extensive user-requirements survey was done and mathematical models, rules and algorithms were applied (Konstantinos G. Zografos, 2002) to develop Incident response logistics (IRL) system for real-time decision making in Traffic management. The goal was to minimize incident response time. NFRs addressed in the said work were Real time-a) dispatching of response units (RUs), b) routing of the RUs, and c) on-scene management and d) Incident evolution monitoring.

Formal Design Analysis Framework (FDAF) was used (Cooper K., Dai L., Deng Y., 2004) to develop real-time decision making system by analyzing non-functional requirements using the example application of Domain Name System (DNS). The proposed solution availed the aspect-oriented UML based approach. NFRs addressed were response time and resource utilization. The output indicated the presence of overloaded components in the system design. If the overloaded component exists, then the UML architecture is refined with each iteration till the clients' requirements are met.

The application of Real time decision making is highlighted (Abdunabi T., 2014) by assessing the challenges in multi assets trading in the financial Industry. The Holonic Intelligent Multi-Agent Algorithmic Trading System (HIMAATS) proposed in this paper addresses the issues of timely decisions, especially, for high-frequency trading. The proposed work addresses the software functional requirements (multi-asset, multi-strategy, real-time risk assessment, etc.), and non-functional requirements (autonomy, high-throughput, low-latency, modularity, scalability, etc.). The authors try to address jobs of traders to trade different types of assets, and provide real-time risk assessments, status, and performance of the diversified portfolio.

The author (Santos E., 2012) has worked on developing a real-time and adaptable framework based on business process configuration determined by non-functional requirements for Airplane Check-in and Boarding process. Considering the variability of a process, the proposed model is supposed to select context specific business process model.

An analytical model is developed to accept low quality real-time sensory multivariate contextual information to deliver progressive data streams of higher quality (Anagnostopoulos, 2016). The goal is to get good quality data for knowledge extraction and the task of predictive analysis. The evaluation methods used were experimental evaluation and comparative assessment with quality-unaware methods.

2.2.2 RTDM and AI

Real-time decision making can be thought of as a combination of three congregating areas: IOT, decision making, and real-time (TIEN, 2019). Training-based decision making (DM) using artificial intelligence makes use of processing, reacting and learning. DM in connection with real-time tools include sensing (i.e., taking input of live data), processing (i.e., application of data analytics with real-time information), reacting (i.e., responding with best choice of solution in real-time), and learning (i.e., applying deep neural networks). A detailed discussion on the real-time decision making by Simulation-based approach in manufacturing automation systems can be found in (Hyun Joong Yoon, 2006). It states that maximum of the latest researches in particular area, focus on knowledge-based approaches as they offer quicker and greater powerful surroundings for RTDM. An application of RTDM in the oil and gas industry proposed by (Erik Bek-Pedersen, Morten Lind, & Bjarne André Asheim, AI Based Real-Time Decision Making, 2019) is based on proposing solution to provide real-time decision support for control room operations to reduce human actions in terms of operator decisions by avoiding abnormal events from escalating and cause plant upsets. The proposed model is based on Multilevel Flow Modeling (MFM), an Artificial Intelligence technology which combines complex physical systems and cognitive science. The work uses AI tools like ontology and rule-based reasoning.

In certain situations decision needs to be taken dynamically by taking action based on latest data. A post-disaster route finding Web Application (Shaohan Hu, 2015) is developed using AJAX and NLP processor. The work provides options to choose among multiple alternatives to take decisions based on run time data. Timely data is retrieved under two constraints defined by the authors. One constraint is decision to be taken under certain time limit called as decision deadline. Another constraint is about data to be used for taking action. In this case the data retrieved is as latest as possible over a certain time range. If the range is increased then the freshness of data may get hampered.

Intelligent decision-making of human resource allocation is studied using AI (He Ma & Jun Wang, 2021). This work makes use of case study method and the data analysis method for real-time monitoring of HR configurations and improves HR management work efficiency.

2.2.3 Metrics which define real-time system

While considering the software performance measurement of a real-time system, it is mandatory to study the types of metrics and their meaning (Association Forum). The list of metrics used to examine the performance of a real-time IT system is as below-

Performance profiling: It is a process used to identify that part of code which is responsible for poor performance of software. This helps to identify areas and create scope for code optimization.

A-B timing: A-B timing is a measure of time calculation to take the control from one given point in code to another. Thus it is an assessment measure of responsiveness of the real-time code within its own context.

Response to external events: In real-time systems, response time is very important. This time is a measure of finding the time lag between the occurrence of an external event and the time of response by software to that event.

RTOS task performance: This includes the time required for a task to reach its completion in multi-tasking OS once the event is triggered. It also includes task profiling performance measurements on a task-by-task basis.

2.2.4 Applications of real time decision making systems and supporting NFR

During the literature review the various applications of real-time decision making systems and corresponding NFRs addressed are listed in table below-

Authors	Application	Characteristics of non-Functional Quality Attributes addressed
(Konstantinos G. Zografos, 2002)	Incident response logistics (IRL) in Traffic management	Real time- a. dispatching of response units (RUs), b. routing of the RUs, c. on-scene management, d. Incident evolution monitoring
(Chung L., 2003)	home appliance control system (HACS)	Adaptability
(Cooper K., Dai L., Deng Y., 2004)	Domain Name System (DNS)	response time and resource utilization
(Santos E., 2012)	Airplane Check-in and Boarding process	Performance, Throughput, Response Time, Reliability, Fault tolerance, Accuracy, Availability
(Gutierrez F., 2012)	Partial Virtual Communities (Education management System)	performance, uptime (availability), maintainability and scalability, response time, extensibility, privacy, security, usability
(Belaggoun, 2013)	Decision-Making for Self-Adaptive Systems	Ex. 1-The Vacuum Cleaner : avoid tripping hazard, minimize energy costs Ex. 2- Remote Data Mirroring: Minimize Operational Expense, Maximize Data Reliability and Maximize Network Performance
(Ruiz-López T., 2013)	An health care system, Positioning Service of a real system.	dynamicity, adaptation, Context-awareness, Heterogeneity of technological solutions., Adaptivity and personalization, Prioritization changes
(David Ameller and Claudia Ayala, 2013)	architectural decision making	Elicitation, documentation, and validation.
(Abdunabi T., 2014)	Financial market	autonomy, high-throughput, low-latency, modularity, scalability
(Saadatmand M., 2015)	Non- Functional Requirements (NFRs) in designing systems	Example application: Mobile Phone Camera picture quality, Battery life, and flash
(O. N. Dolinina, 2019)	gas turbines faults	usability; understandability; learnability;

Table1: Applications of real time decision making systems and supporting NFR during 2002-2021

3. REVIEW OF LITERATURE

Performance analysis gives a broad picture of dynamic behavior of a system and the prediction of performance measures such as its throughput, utilization or response time. (Dušan Okanović, 2019) States that current ways of performance analysis is in terms of dashboards which are simply data driven. No explanation is provided in natural language, so it limits the understandability of the user. Author proposes a method that takes into consideration the user concerns. The approach generates natural language description as a output which is an answer to the user concern. Authors have developed prototype implementation of the approach using vizard tool. As a future work, the author plans to 1. collect the information requirements of non-experts as well as experts and use the results to refine the reports. 2. study different use cases and their ways to accept user concerns and report the results. 3. Develop a

model that will incorporate metrics and other data from different other tools. Web sites performance was further studied for scaling the web application by (Pankaj N. Moharil, 2021) who worked on scaling the Web Application with more workload and more users. The authors propose an algorithm to improve load balancing. It tries to determine the wide variety of customers of the Web utility with the maximum intense wide variety of parallel and simultaneous customers with a variable load. The author tries to maintain best possible balance among maximum possible number of users, variable workload and the optimal performance for a web application. The authors make use of NeoLoad automated performance tool to carry out the experiments. The results show that addition of more resources is required than to make modifications to the application, in order to make the application more scalable.

The problem of un-availability of methods to predict performance related events in multi-system environment using machine learning was tried to solve by (Andreas Schörghener, 2019). The methodology uses supervised learning method to train the model by observing real-time data of 90 systems over the span of 20 days. The said work is an application of proactive fault management. Gaming applications are the most widely used softwares by youngsters as well as elders. The game Minecraft was explored for their performances by (Jerom van der Sar, 2019) where the authors state that currently no benchmark exist to analyze the performance and scalability of the gaming application Minecraft. The author designs and uses the Yardstick benchmark to examine the overall performance of gaming services. Performance and scalability are the performance metrics considered in this work. As a future work, the authors propose to design new scalability techniques and explore new performance metrics for other gaming services.

(Hui, 2020) have attempted to integrate positioning system and AI; and used it for recognition of the student in online English teaching and tested the model for performance. The authors have worked to improve the performance of online teaching in the English language. To monitor and identify students' status, the authors of the said paper have developed an AI based model. The system's recognition effect is evaluated using performance testing methods. The system makes use of positioning method based on an improved deep belief network for real-time position control.

Agile software development (ASD) was widely studied for its performance during its execution by various practitioners as well as researchers. (WEICHBROTH, 2021) state that agile software development (ASD) mainly focus on feature driven incremental development without need of documentation and tend to neglect NFRs. Objective of the work is to perceive and check out the practices utilized in industrial ASD initiatives to perceive (identify), elicit and report (document) NFRs. As part of methodology, the writer used outcome of SLR to design interview plan to conduct interviews with enterprise experts. The SLR found out some of techniques associated with the timing of NFR identity in addition to 13 elicitation practices and 20 documentation techniques. As part of output, authors observed that practitioners' try and begin figuring out NFRs early in the project or with non-stop refinement with the every iteration of the project. Different choices about documentation techniques were found like user stories, acceptance criteria, definition of Done, etc. Some of the elicitation practices included prototyping, brainstorming, elicitation workshops, etc. The database of articles used for SLR was taken from Scopus. In future work author proposes to define a NFR management framework for ASD by evaluating effectiveness of the recognized NFR elicitation and documentation methods and approaches. Also the authors propose to find out the parameters related to NFR approaches originating from the implementation of agile methods across software development organizations.

Solutions are provided for web application performance issues by many authors. (S. Pradeep, 2019) has presented the important software testing tools which are mainly used for stress and performance testing of the web applications. The author has given detailer overview of various tools for Load Testing, Performance Testing, Scalability Testing, etc. Demonstration was shown for popular performance testing jobs like Generation of HTTP Requests for Generation of Load in Jmeter, Specifying Threads (Users) in Apache JMeter, The View of locustfile.py in Locust, etc. Similarly, (Fernando Maila-Maila, 2019) has done a case study of a web application to carry out a performance test. The complete life cycle of a quality testing is performed. Quality measures considered for performance characteristics were Time Behavior, Resource Utilization, and Capacity.

(Dheeraj Chahal, 2019) considers the scenario of batch execution of processes with dependencies. The current techniques consider existing cores, size of memory, priority or burst time of jobs. The author provides a simulation model that is based on resource contention models (conflict over access to a shared resource) in a concurrent run. Optimal mapping of jobs onto available nodes ensures minimum total execution time. Dheeraj Chahal et al have limited their experiments to a cluster of 4 servers only however this approach can be used with massive clusters with the aid of simultaneous execution of the worker thread to reduce the simulation time. In future, Dheeraj Chahal et al intend to embed the simulator in his job scheduler in order that a task is right now located at the server while suitable server for its execution is decided through the simulator.

(Robert Nishihara, 2017) provided a solution for challenges posed by distributed execution frameworks w.r.t. performance requirements latency and throughput using techniques like ML and reinforcement learning. They used the tools like Real time Simulation, Tools: Spark, MPI, and TensorFlow.

RTDM methods are meant to respond to highly uncertain and complex nature of the intelligent production process (Liping Zhang, 2021). The proposed approach defines a framework that is used to achieve real-time decision making using machine learning techniques.

(Yue Zhu, 2020) has used machine learning algorithm for the cloud platform. The author has found that the current cloud platform performance benchmark cannot test the machine learning abilities of cloud platforms. The author has proposed an overall performance assessment approach for a cloud platform with machine learning techniques. Limitation of the proposed work is that, the only metric selected was load time. In future work author is going to introduce new metrics, such as the PR curve with better accuracy and recall rate.

Paper#	key1	key2	key3	Methodology
(Pankaj Moharil, 2021)	N. Web application	Workload	Scalability	Algorithm design
(Yue Zhu, 2020)	Online teaching	positioning	AI	model-driven software development method
(Hui, 2020)	Agile	NFR identification, elicitation and documentation	interview and SLR	SLR and interviews
(Dušan Okanović, 2019)	Web application	Concern driven	vizard tool	Experimentation and prototype implementation
(Andreas Schörgenhuber, 2019)	performance-related events prediction	multiple systems	Supervised Machine Learning	model building by experiments
(Jerom van der Sar, 2019)	performance and scalability	minecraft (gaming)	Benchmark	operational model and real-world experiments
(Yue Zhu, 2020)	Cloud	machine learning application	performance	Design of evaluation method
(Hui, 2020)	Web application	performance testing	Tools	Comparative evaluation using various performance testing tools
(Dheeraj Chahal, 2019)	minimize completion time	batch jobs with dependencies	job scheduling algorithms	Simulation
(Fernando Maila-Maila, 2019)	Web application	Open Source Software	case study application	LR, experimentation in Jmeter, method of generating data by Weighted matrix of features of software products, Analysis of the data by Quantitative analysis

Table 2: Work done related to Software systems and NFR during 2019-2021

*In table 2, key refers to the key element on which the author has worked upon

The run-time uncertainties causing degradation of quality of service (QOS) are handled by a ML based decision making tool (Henry Muccini, 2019). The tool aids architectural adaptations to recognize the requirement for revision in the beginning stages of system development. The developers are also able to view real-time QOS data, adaptation process in almost real-time and mention required configurations. Blockchain technology is used for ensuring security of the proposed framework.

The authors (Schahin Tofangchi, 2017), in their multidisciplinary research, have worked upon improving efficiency and quality of decision making in the domain of healthcare practices. Using healthcare analytics in clinical practice the authors developed a ML-based real-time DSS to find out possible best treatments and communicate its prediction confidence as well as patient attributes like Height, Tumor type, Gender, etc.; which are necessary for taking run-time decisions. Further in the domain of healthcare, real-time processing is studied by the use of IOT, Fog computing and Machine learning (Anwasha Banerjee, 2020). The proposed idea is implemented for ICU patients by monitoring and collecting their physiological movements by IOT devices. The collected data is sent to FOD nodes where ML techniques are used for training the model during first 3 hours after the patient is admitted. As the model is trained, further real-time data being received from Fog node is categorized based on whether the node needs attention by the patient or not.

4. DISCUSSION

We have analyzed research papers from Google scholar and Scopus database. After analyzing the research papers based on performance of various IT systems and performance engineering of real-time decision making systems, the following research gaps have been identified-

- Existing literature does not have comprehensive architecture for RTDM IT systems. The reason being that the existing architectures do not address all the RTDM capabilities.
- Comprehensive Quality Engineering framework for RTDM is currently not available in the existing literature where the NFRs like Performance profiling, A-B timing, Response to external events, RTOS task performance are not covered.
- Very few existing RTDM architecture have ML/DL methodologies incorporated for quality engineering framework.
- We found no existing RTDM architecture to incorporate ML/DL methodologies for Chaos engineering to make the system more resilient to failures.

Based on the research gaps discussed, the following are the research questions of relevance:

- How can I improve the performance of real-time decision making?
- Which NFR criteria can produce best performance for a RTDM system?
- How can quality Engineering aid real-time decision making?

5. CONCLUSION AND FUTURE SCOPE

Though extensive research has been done, a full-fledged criterion of the software quality controlling is still explored on a comparatively superficial level [5]. Based on the analysis given in sections 2, the categories on which people have worked upon is derived as given in table below-

Category	Number of research papers
ML-Quality Engineering	6
ML-RTDM	4
ML-RTDM- Quality Engineering	1
DL- RTDM	1
DL-Quality Engineering	0
DL-Real time	0
DL-Decision making	0
Quality Engineering-RTDM	8
Quality Engineering-Decision making	6
Quality Engineering-AI	2
RTDM-AI	1
DM-Quality Engineering	1
DM-AI	4
Quality and statistical modeling	1
RT and ANN	1
Quality Engineering-RTDM -Deep learning	0

Table 3: Broad categories of work on which people have done research

The systematic literature review in this proposal reveals that people have not worked on quality parameters of real-time decision making system's using Machine learning and Deep learning. Given this fact we would be working on designing a machine learning/deep learning based quality engineering framework for real-time decision making IT system. Based on our review we have identified our further objectives as to,

- find shortcomings in existing RTDM for specific IT systems,
- design specific Machine learning and Deep learning algorithms that will support in enhancing the real-time decision making systems
- create a quality engineering architecture/framework for specific RTDM in IT systems.
- identify public databases to test the framework.
- apply this framework in specific application in the IT domain.
- identify the specific metrics to test the quality engineering framework.

REFERENCES

- [1] (n.d.). Retrieved September 7, 2021, from Association Forum: <https://www.associationforum.org/mainsite/browse/professional-practice-statements/performance-measurement-metrics>
- [2] (n.d.). Retrieved September 23, 2021, from 8allocate: <https://8allocate.com/article/top-8-tools-and-libraries-for-ml-and-dl-project-development/>
- [3] Abdunabi T., B. O. (2014). Holonic Intelligent Multi-Agent Algorithmic Trading System (HIMAATS). *International Journal of Computers and their Applications*, 21(1), 54-61.
- [4] Alice Mugisha, Victoria Nankabirwa, Thorkild Tylleskär, & Ankica Babic. (2019). A usability design checklist for Mobile electronic data capturing forms: the validation process. *BMC Medical Informatics and Decision Making*, 1-11.
- [5] Anagnostopoulos, C. (2016). Quality-optimized predictive analytics. Springer, 1034-1046.
- [6] Andreas Schörgenhuber, M. K. (2019). Can we Predict Performance Events with Time Series Data from Monitoring Multiple Systems. *ACM*, 9-12.
- [7] Anwesha Banerjee, B. K. (2020). A Secure IoT-Fog Enabled Smart Decision Making system using Machine Learning for Intensive Care unit. *International Conference on Artificial Intelligence and Signal Processing (AISP)*, (pp. 1-6).
- [8] Bahar Farahani, Mojtaba Barzegari, & Fereidoon Shams Aliee. (2019). Towards Collaborative Machine Learning Driven Healthcare Internet of Things. *COINS* (pp. 1-7). Crete, Greece: ACM.
- [9] Belaggoun, N. B. (2013). Supporting Decision-Making for Self-Adaptive Systems: From Goal Models to Dynamic Decision Networks. Springer International Publishing, 221-236.
- [10] Berna Unver, Özgür Kabak, Y. Ilker Topcu, Armagan Altinisk, & Ozcan Cavusoglu. (2020). A decision support system for proactive failure prevention: a case in a leading automotive company. *Journal of Enterprise Information*, 1-36.
- [11] Chung L., C. K. (2003). Developing adaptable software architectures using design patterns: An NFR approach. *Computer Standards and Interfaces*, 25(3), 253-260.
- [12] Cody Baker, Lin Deng, Suranjan Chakraborty, & Suranjan Chakraborty. (2019). Automatic Multi-Class Non-Functional Software Requirements Classification Using Neural Networks. *IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC)*, (pp. 610-615).
- [13] Cooper K., Dai L., Deng Y. (2004). Performance modeling and analysis of software architectures: An aspect-oriented UML based approach. *Proceedings of the International Conference on Software Engineering Research and Practice, SERP*, (pp. 111-117).
- [14] David Ameller and Claudia Ayala, J. C. (2013). Non-functional Requirements in Architectural Decision Making. *IEEE Software*, 30(2), 61-67.
- [15] Dawei Han, T. K. (2007). Uncertainties in real-time flood forecasting with neural networks. *Wiley InterScience*, 223-228.
- [16] Dheeraj Chahal, B. M. (2019). Simulation Based Job Scheduling Optimization for Batch Workloads. *ACM*, 313-320.
- [17] Dušan Okanović, A. v. (2019). Concern-driven Reporting of Software Performance Analysis Results. *ACM*, 1-4.
- [18] Erik Bek-Pedersen, D. H., Morten Lind, T. U., & Bjarne André Asheim, K. T. (2019). AI Based Real-Time Decision Making. Abu Dhabi International Petroleum Exhibition & Conference. Abu Dhabi, UAE.
- [19] Erik Bek-Pedersen, D. H., Morten Lind, T. U., & Bjarne André Asheim, K. T. (2019). AI Based Real-Time Decision Making. Abu Dhabi International Petroleum Exhibition & Conference held in Abu Dhabi, UAE.
- [20] Fernando Maila-Maila, M. I.-P. (2019). Evaluation of Open Source Software. Springer Nature Switzerland AG 2019, 75–82.
- [21] Fikri Dweiri, S. K. (2016). Designing an integrated AHP based decision support system for supplier selection in automotive industry. *Expert Systems With Applications* www.elsevier.com/locate/eswa, (pp. 273-283).
- [22] Gutierrez F., B. N. (2012). Designing the software support for partially virtual communities. *Collaboration and Technology*, 18th International Conference, CRIWG 2012, Raesfeld, Germany, (pp. 70-88).
- [23] He Ma, & Jun Wang. (2021). Springer Nature Switzerland, 201-207.
- [24] Henry Muccini, K. V. (2019). ArchLearner: Leveraging Machine-learning Techniques for Proactive Architectural Adaptation. *ECSA* (pp. 1-4). Paris, France: ACM.
- [25] Hillary, N. (2005). Measuring Performance for Real-Time Systems. Retrieved August 26, 2021, from <https://www.nxp.com/docs/en/white-paper/CWPERFORMWP.pdf>: <https://www.nxp.com/docs/en/white-paper/CWPERFORMWP.pdf>
- [26] Hui, L. A. (2020). An artificial intelligence recognition model for English online teaching. *Journal of Intelligent & Fuzzy Systems*, 40(2), 3547-3558.
- [27] Hyun Joong Yoon, W. S. (2006). Simulation-based real-time decision making for manufacturing automation systems: a review. *Int. J. Manufacturing Technology and Management*, 8, 188-202.
- [28] Jerom van der Sar, J. D. (2019). Yardstick: A Benchmark for Minecraft-like Services. *ACM*, 242-252.
- [29] Konstantinos G. Zografos, k. N. (2002). A real-time decision support system for roadway network incident response logistics. *elsevier*, 1-18.
- [30] Liping Zhang, Y. H. (2021). Data-Driven Dispatching Rules Mining and Real-Time Decision-Making Methodology in Intelligent Manufacturing Shop Floor with Uncertainty. *MDPI*, 1-23.
- [31] Martin, M. (2021, August 27). Top 10 BEST Decision Making Tools for Business in 2021. Retrieved September 23, 2021, from <https://www.guru99.com/decision-making-tools.html>
- [32] Max Scheerer, Axel Busch, & Anne Koziolk. (2017). Automatic Evaluation of Complex Design Decisions in Component-based Software Architectures. *MEMOCODE* (pp. 1-10). Vienna, Austria: ACM.
- [33] Naina Handa, A. S. (2020). An Inclusive Study of Several Machine Learning Based Non-functional Requirements Prediction Techniques. Springer Nature Singapore Pte Ltd. 2020, (pp. 482-493).
- [34] Naina Handa, Anil Sharma, & Amardeep Gupta. (2020). An Inclusive Study of Several Machine Learning Based Non-functional Requirements Prediction Techniques. Springer Nature Singapore Pte Ltd., 482-493.
- [35] O. N. Dolinina, V. A. (2019). The Way of Quality Management of the Decision Making Software Systems Development. Springer International Publishing, 763, 90-98.
- [36] Pankaj N. Moharil, S. J. (2021). Enhancement in Algorithmic Design for Scalability Measurement of Web Application. *Advances in Computational Intelligence and Communication*, 251-263.
- [37] Pawar, S. H., & Thakore, D. (2017). An Assessment Model to Evaluate Quality Attributes In Big Data Quality. *International Journal of Computer Science Trends and Technology (IJCTST)*, 5(2), 373-376.
- [38] Pooley, A., & Chan, J. H. (2021). A case study on performance assessment framework A case study on performance assessment framework. *Int. J. Business Performance Management*, 22(1), 1-14.
- [39] QA Infotech. (2021, March 11). Best 9 Performance Testing Tools Trending In 2021. Retrieved September 23, 2021, from <https://qainfotech.com/best-9-performance-testing-tools/>
- [40] Rakesh Ranjan Kumar, & Chiranjeev Kumar. (2018). A Multi Criteria Decision Making Method for Cloud Service Selection and Ranking. *International Journal of Ambient Computing and Intelligence*, 9(3).
- [41] Robert Nishihara, P. M. (2017). Real-Time Machine Learning: The Missing Pieces. 16th Workshop on Hot Topics in Operating Systems, (pp. 106-110).

- [42] Ruiz-López T., N. M. (2013). REUBI: A Requirements Engineering method for ubiquitous systems. *Science of Computer Programming*, 78(10), 1895-1911.
- [43] S. Pradeep, Y. K. (2019). A Pragmatic Evaluation of Stress and Performance Testing Technologies for Web Based Applications. *IEEE Xplore*, 399-403.
- [44] Saadatmand M., T. S. (2015). A Fuzzy Decision Support Approach for Model-Based Tradeoff Analysis of Non-functional Requirements. *Proceedings - 12th International Conference on Information Technology: New Generations, ITNG 2015*, (pp. 112-121).
- [45] Sahu, N., Sahu, A., & Sahu, A. (2018). Cluster approach integrating weighted geometric aggregation operator to appraise industrial robot. *Emerald Insight*, 47(3), 487-524.
- [46] Santos E., P. J. (2012). On the dynamic configuration of business process models. *Enterprise, Business-Process*, (pp. 331-346).
- [47] Schahin Tofangchi, F. B. (2017). Distributed Cognitive Expert Systems in Cancer Data Analytics: A Decision Support System for Oral and Maxillofacial Surgery. *Thirty Eighth International Conference on Information Systems, South Korea 2017*, (pp. 1-22).
- [48] Shaohan Hu, S. Y. (2015). Data Acquisition for Real-time Decision-making. *IEEE Real-Time Systems Symposium*, 185-194.
- [49] Tien, J. M. (2017). THE SPUTNIK OF SERVGOODS: AUTONOMOUS VEHICLES. *Systems Engineering Society of China and Springer-Verlag Berlin Heidelberg 2017*, 1-30.
- [50] TIEN, J. M. (2019). Convergence to real-time decision making. *Front. Eng. Manag. Higher Education Press 2019*, 1-19.
- [51] Turban, A. L. (2007). *Decision Support System And Intelligent Systems* (7 ed.).
- [52] Unver, B. I. (2018). An AHP based prioritization model for risk evaluation factors in the automotive industry. *International Journal of Analytic Hierarchy Process*, 10(1), 83-97.
- [53] WEICHBROTH, A. J. (2021). A Qualitative Study on Non-Functional. *IEEE Access*, 9, 1-18.
- [54] Yue Zhu, S. Y. (2020). A Performance Evaluation Method for Machine Learning Cloud. *Springer Nature Singapore Pte Ltd. 2020*, 249-259.

