Journal of Artificial Intelligence and Computational Technology

Journal homepage: https://ojs.omgfzc.com/index.php/JECT

Machine Learning: A survey of requirements prioritization: A Review Study

*Kawthar Ishag Ali Fadlallah**, Moawia Elfaki Yahia Eldow

[1]Department of Computer Science, Omdurman Islamic University, Omdurman, Sudan

[2]Faculty of Mathematical Science, University of Khartoum, Khartoum 11115, Sudan, University of North Texas, Denton, TX, USA

*Correspondence: E-mail: kawthar1140@gmail.com

Article Info

Article History:
Submitted/Received 5 July 2024
Revised in revised format 10 August 2024
Accepted 18 September 2024
Available 15 October 2024
Publication Date 15 October 2024

Keyword:
Requirements Prioritization, Machine Learning, Dependency,Scalability

Cite this article: *Fadlallah, K. I. A., & Eldow, M. E. Y. (2024). Machine learning: A survey of requirements prioritization: A review study. Journal of Artificial Intelligence and Computational Technology, 1(1).*

ABSTRACT

In any software systems, the requirements prioritization considered as pivotal task. This paper aims to explain and discuss the works done on requirements prioritization based Machine learning along with the dependency-aware requirements prioritization. Machine learning has become of attention to scientists, researchers, and users because of the existence of vast data and deep learning algorithms that can analyze massive sets of data. The basic algorithms are used for dependency learning calculation, resolve the stakeholder's conflicts, classify requirements, and scalability improvement. This paper will present a brief background and comprehensive presentation of a number of machine learning techniques for requirements prioritization and those concerning the requirement dependency in its simple, complex, and Hybrid form. There are a number of papers, articles, and research papers that deal with requirements prioritization, few of them handling the dependency. This paper will present a brief background for several requirements prioritization based on machine learning. then make a comparison and discussion of a number of selected techniques in terms of algorithm type, issues addressed, and evaluated data level. method that handling the dependency regarding of strengths and weakness.

1. Introduction

Requirements Prioritization (RP) defines as a complex decision-making process as it controls the order of implementation and, accordingly, the delivery of a software system. Incorrect RP may lead to software project failures due to exceed budget and schedule in addition to the low-quality product. Numerous aspects influence RP, one of which is handling the requirements dependency. Ignoring the requirements dependencies handling can cause software development failures (Noviyanto et al., 2023). Machine learning (ML) used to automate requirements prioritization via multiple techniques such as PageRank, RankBoos, Case Based method, Supervised Classification Technique, and the most important issue is in this context is handing the dependency between requirements.

We can use machine learning algorithms in requirement prioritization, dependency learning and calculation, resolve the stakeholder's conflicts, and scalability improvement. A lot of prioritization techniques are built in different machine learning algorithms. We will introduce an overview of numerous techniques work in this field, such as the simple, complex, novel, and hybrid. We found that there are several techniques that made a comparison between their own work with existing techniques. When developing the software systems, requirements prioritization has been achieved by many approaches. developers expected to select which algorithms to use in their project after evaluate and compare between algorithms. however, according to researchers in (Noviyanto et al., 2023).
most of existing techniques are still at the research phase and have not been applied to resolve real-world industry problems.

2. Related Works

This section, we will present and associate several papers and researches connecting the requirements prioritization based artificial intelligent(AI) and machine learning, dependency between requirements, stakeholders and developers' collaboration, and scalability issues.

A literature survey provides roadmap and guidelines to academics and practitioners to easily get to explore a particular topic by choosing high-quality articles or reflects that are related, significant, important, and substantial and summarizing them into one complete report. Moreover, it obtains a great starting point for analysts to understand the current state of the art of research area, evaluate and compare unique analysis in that specific region, and it makes sure of that work has not be done before.

It can provide evidences as to where future suggestions for future work in relate summarizations. It gives a helpful examination of the process and methods of other researchers. The related works will discuss in the following section.

2.1. Requirements prioritization based AI and machine learning techniques

(Duan et al., 2009) proposed a requirements prioritizing technique based on stakeholders' business goals, preferences, and cross-cutting issues such as security and performance requirements, applying data mining and machine learning algorithms. The Spherical Kmeans (SPK) clustering technique is used in the implementation phase. The NFR classifier is a data mining tool that discovers and categorizes a variety of non-functional requirements (NFR) like performance, security, and performance. The classifier is based on the idea of weighted indicator phrases, which weighted each possible phrase based on how efficiently it indicated the existence of a specific NFR type. To validate and assess the technique two case studies constructed on the Ice Breaker System's requirements, as well as a gathering of stakeholders' raw feature requests captured from the SugarCRM discussion environment. The finding highlighted the valuable in organizing the huge amount of stakeholders' needs. And therefor it scales to many thousands of requirements. however, could not produce perfect precision or recall in the findings due to the probabilistic nature in data mining and information retrieval techniques. Also the dependency between requirements is ignored.

DOI: https://doi.org/10.70274/jaict.2024.1.1.18.
p- ISSN 34562-3478 e- ISSN 5443-1243

(Duan et al., 20١٢) proposed a machine learning approach that adopts Case-Based Ranking (CBRank) method for requirement prioritization. (CBRank) combines the stakeholders' preferences with requirements ranking estimates. CBRank are empirically assessed on simulated data and compared to a state-of the-art prioritizing approach, validating the ability of method to allow the management of the trade-off between elicitation effort and ranking precision and utilization of domain knowledge. Then these experimental explanations are supplemented by a case study on a real software project. An experiment conducted with 23 participants to compare two tool-supported versions of CBRank and AHP in terms of ease of use, time consumption for completing the job, and accuracy of final ranking. the authors privilege that their approach overtakes AHP in the worst-case scenario. However, it is more sophisticated due to pair sampling polices and the dependency among requirements is not considered.

(Achimugu & Selamat., 20١٥) presented a hybridized requirement prioritization algorithm that adopted differential evolution and the k-means algorithm, to make a better-prioritized strategy to handle scalability, rank reversals, and calculating complexity. The algorithm uses requirement preference weights derived from stakeholder linguistic evaluations. To demonstrate the technique, they used RALIC dataset, which includes requirements with associative weights of stakeholders. The finding state that the proposed approach could handle a large sets of requirements professionally by decreasing conflicts among prioritized requirements. consequently, this could assist software developers to select the most important and least important requirements, which will aid with software release planning and decrease contract, trust, and agreement abuses. However, it negated the dependency among requirements.

(Babar et al., 20١٥) presented an expert system, called the Priority Handler (PHandler) that appoints, to resolve the scalability problem. The proposed framework integrates three approaches, the Value-based Intelligent Requirement Prioritization (VIRP), the Back Propagation Neural Network (BPNN), and the Analytic Hierarchy process (AHP). PHandler creates an initial ranking of requirements by an expert. And then neural networks and the Fuzzy c-means clustering method adopted to establish clusters of requirements according to priority. The final ranking obtained through AHP. The results illustrate that PHandler system improved performance in term of the number of prioritized requirements Particularly, it has capability to handle a dataset of more than 500 requirements. And it achieved 93.89% when compared to existing approaches. It can generally have excellent outcomes in ranking requirements specifically it can outperform CBRank 80% in term of efficiency. Moreover, using the BPNN decreases the specialists' preferences and improves the performance in terms of time. The key challenge of PHandler system is the need of expert business analysts since a robust analyst's knowledge is essential to estimate precise values of requirements classification factors. The main drawbacks PHandler represented in neglects the dependency association among requirements. And the expert systems do not clarify the reason behind taking a decision. Moreover, simply automate complex processes, and lack of supporting tool is detected.

(Gupta & Gupta., 20١٨) proposed a collaborative RP model with two surely understood opinions of stakeholders and developers. It helps in right decision making due to re-prioritize or improve the priorities as given by stakeholders and developers, which associated with risk estimation for each requirement. The model consists of two phases; the first one is training and selection, in which both stakeholders and developers will be trained to share common opinion about the priorities of the requirements. The second phase is score computation and clustering; in which, the difference between the ranks and the total difference in requirements rank for each person will calculated using formulas. Then the results are investigated and regularized for the purpose of clustering using k-means clustering algorithm and further classification. The model has been supported with an example case study and by a survey that is directed to IT companies. The result demonstrated that the response for the applicability of the proposed model from the industry is significant. As it can help in reducing the disagreements between stakeholders and developers. Also model could

DOI: https://doi.org/10.70274/jaict.2024.1.1.18.
p- ISSN 34562-3478 e- ISSN 5443-1243

reduce the over all development time. However, the authors argue that the formula used to calculate the percentage of the role of stakeholders and developers in the project is subjective in nature. This due to the fact that the formula is suggested by the project manager which may lead to positive biasness. Additionally, the unawareness of dependency between requirements is also noticed.

(Singh et al., 20١٨) proposed a hybrid model, called 'ANN fuzzy AHP model, which constructed based on assessment of seven requirements prioritization techniques. They documented numerous limitations associated to these techniques such as scalability and most of the techniques make multiple optimal outcomes which lead solid inconsistency between fuzzy decision. Thus, this model aimed to overcome these limitations and drawbacks. The initial priority of elicited requirements stored in the database based on the experts' criteria. Then the data are prepared and analyzed in the form of a FAHP and ANN respectively, the information enters to an assembled program of MATLAB software by way of fuzzy pair-wise comparisons matrices. The model was validated on the case study of the supplier selection problem. The outcome highlighted a good result obtained in terms of advanced priority as compared to existing techniques However, it ignores the dependency between requirements.

(Sandanasamy & Selvi., 20١٩) proposed a novel quality based software requirements prioritization method using Takagi Sugeno neuro fuzzy logic. The requirements are collected from numerous stakeholders based on some criteria specifically stakeholder's importance (SI), cost, time and risk which are provided as the input. The initial step in implementing Takagi Sugeno inference is to define the input and output parameters. The input parameters for the four mentioned input are recognized as LOW, MEDIUM, HIGH, and the output parameters priority are recognized as VERY LOW, LOW, MEDIUM, HIGH, VERY HIGH. Then A fuzzy rule is IF-THEN rule with condition and action should be definite to perform the output function. it was tested through FisPro toolkit. The outcome shows that as the stakeholder importance is high, cost, time and risk are low, the priority is high. However, the dependency among requirements is neglected and resolving conflict between stakeholders is ignored.

(Singh et al., 20١9) proposed a novel hybrid model, which combines logarithmic fuzzy trapezoidal approach (LFTA) with artificial neural network (ANN). This model aimed to resolve conflicts regarding which one of the requirements must be executed first, as per their weight priority in multi-criteria decision making (MCDM). This model is beginning initial priority given by the experts to each requirement in the database. Afterward, decision makers (DM) get their requirement selection criteria for prioritization. Then data information drives as fuzzy pairwise comparison examinations into a pre-accumulated MATLAB programming. Additionally, the ready ANN (artificial neural network) tests the correctness and inconsistency of decision makers. The mode was evaluated using case study on real-life example of college selection (CS) to select the best college according to three principles (faculty profile, college infrastructure and technical, cultural activity). The results indicate that the proposed model outperformed existing techniques like Extent Analysis (EA) or Fuzzy Preference Approach (FPA) and FAHP in team of accuracy. However, it suffers from several drawbacks such as dependency ignorance, complexity and lack of supporting tool.

(Sher et al., 20١9) proposed a multi-aspects requirements prioritization technique for value based system VBS systems., where both technical aspects and business aspect are considered. Fuzzy C-Means (FCM) was used as clustering technique to take the most critical cluster of requirements to be included the release planning. The input in this technique is created on weights allocated to each requirement by diverse experts. The input divided into two types which are business and technical aspect values and fuzzy c-means parameters. The output is computed values for both types of aspects associate to each requirement, which is stored in in a database or in a text file. To evaluate the technique, they used Dataset related to enterprise resource planning (ERP) system. Excellent result obtained in handling large scale requirements datasets. However, the dependency between requirements is ignored and the absent of tool support.

(Hafeez et al., 2020) proposed RP framework for multi-stakeholder based prioritization (TMCRP). (TMCRP) utilizes text mining and clustering technique. Hierarchical algorithm clustering is used to assemble the similar attributes of stakeholders into groups based on functionality for example, Development, Finance, Management, and Usage. based on the means of stakeholder's rank, the

DOI: https://doi.org/10.70274/jaict.2024.1.1.18.
p- ISSN 34562-3478 e- ISSN 5443-1243

hierarchical tree which is a two-dimensional figure that depicts accumulation of clusters using rating expressive the weight connection. (TMCRP) could void incompleteness in requirements and difference conflicts between development teams and stakeholders. Moreover, it improves requirement prioritization process in term of accurate features extraction and requirement prioritization in multi-stakeholder situation. The outcomes depict that TMCRP has efficiency outperformed the traditional techniques such as AHP and Winger in handling a large scale of requirements. However, the unawareness of the dependency among requirements is remarked.

(Hujainah et al., 2021) proposed a new semi-automated scalable prioritization model named, SRPTackle, and automation implementation tool (SRPTackle-Tool). SRPTackle utilizes weighted sum model as multi-criteria decision-making method, combine with K-means and K-means++ as clustering algorithms, and a binary search tree. SRPTackle developed to handle the main challenges of the prioritization process such as scalability, time consumption, restricted dependence on expert participation, and lack of automation. The efficiency of SRPTackle is measured through established seven experiments using the RALIC benchmark dataset of a large actual software project. Experiment outcomes expose that SRPTackle able to get 93.0% and 94.65% as least and high accuracy percentages, respectively. The outcomes also highlight the ability of SRPTackle to prioritize large-scale requirements with minimum computation time, and its increase efficiency when compared with other techniques. However, the dependency between requirements is negated.

(Gambo et al., 2021) presented a hybrid mathematical Ranking Model, that Combines the Case Based Ranking (CB Ranking) and the Measurement to Attractiveness by Categorical Evaluation Technique (MACBETH). In this model, once the requirements are captured, firstly the requirements ranked according to the stakeholders' opinions based on the relative importance criteria. Then role of CB Ranking is to provide acceptable support in collaboration between stakeholders. While, (MACBETH)is a technique that handle fuzziness in multi-criteria decision making. The model evaluated using a real-life case study collected 600 requirements at the Centre Hospitalier Department. The result demonstrates that the model ranks a large scale of requirements, and facilitates release planning and support decision-making quality. Moreover, the result shows excellent accuracy of 90% with a response time of0. However, it limited to FRs and ignore the dependency between requirements.

(Sadiq & Devi., 2021) proposed a method using rough set theory to calculate the ranking values of the software requirements (SRs). rough set theory adopted to tactile the subjectivity nature in fuzzy based methods, which may affect the requirements ranking. The method consists of four steps, in the first one identifies the Stakeholders and their requirements. The second, exact opinions of the decision makers during the valuation of FRs and NFRs. The third, represents the relationship among FRs and NFRs using rough numbers. The fourth Compute the ranking values of the FRs based on rough numbers. In particular, the applicability of the method is verified by using an examination system. The authors established that it could captures the exact opinion of the decision makers. Some limitations of this method include limited to small scale of requirements, ignore dependency, and lack of supporting tool.

(Rizawanti et al., 2022) proposed a method called MCBRank, which combines popular MoSCoW method and Case-Based Ranking method to improve the prioritization correctness. Initially, all requirements which originate from multiple stakeholders are listed. Then the important stakeholders are essential to categorize each requirement based on the modified MoSCoW method on five points scale. The MoSCoW organization of M (Must have), S (Should have), C (Could have), and W (Would have) are allocated with numbers as listed below. 'Must not have this' is added to the scale to permit stakeholders to specify the requirements they do not need to be realized. Following, within the classification, each requirement will be ordered using ordinal numbers. To evaluate the MCBRank the e-library system was implemented, where the participants were to prioritize the applicant requirements using the MCBRank method. Good result has been achieved in terms of enhances the importance of ranking correctness. However, the dependency among requirements is not taken into account and the absent of supporting tool is also noticed.

DOI: https://doi.org/10.70274/jaict.2024.1.1.18.
p- ISSN 34562-3478 e- ISSN 5443-1243

(Chua et al., 2022) proposed semi-automated framework for requirements prioritization named (SARiP), which aimed to automate the activities in software requirements prioritization (SRP) process. The proposed SARiP focusses on the parts related to prediction of requirements priority group and ranks requirements using classification tree and ranking algorithm. The SARiP framework start with data pre-processing and analysis of elicited data. Then a requirements list will be supplied. Consequently, the requirements list will be prioritized using the SARiP framework. The implementation of SARiP framework contains two prioritization phases manual and automatic. In the manual phase, the requirements prioritized manually by the project team and stakeholders The manual process using MoSCoW technique, numerical assignment technique and Kano model. The output of the manual phase represents an initial prioritization list, which used as an input for automatic prioritization. The automatic phase ranks the requirements using classification tree and ranking algorithm. Finally, the SARiP framework has been well evaluated in the government sector as case study. However, the authors state that the SARiP does not store the requirements prioritization results in the database. Additionally, the traceability to trace the requirements changes not considered. Further, it has limitations regarding the subjective use of ordinal scales and rankings, and it ignore the dependency among requirements.

(Devadas & Cholli., 2022) proposed a Pugh Decision-based Trapezoidal Fuzzy Requirement Selection model and Gradient Reinforce Learning (PTF-GRL). the (PTF-GRL) model main objective is to address the uncertainty in the opinions between different stakeholders in prioritizing requirements for large scale software projects. The model input is the functional and non-functional requirements of the consistent stakeholders. With the support of Trapezoidal Fuzzy Inference, the qualitative features are mapped with the consistent numeric factors, which maximizes the computational efficiency. Performance is investigated based on four factors: The first factor is accuracy the method displayed improvement of 4%, 7% and 3% compared to JRD-SCRUM, IFS and SRPTackle respectively. The second factor is prioritization time and found that it had decreased time of 30%, 37% and 39% compared with existing methods. The third factor is precision and it was found that our method improves precision by 6%, 10% and 5% associated with the other two methods. The final factor is the test suite execution and method painted improvement of 12%, 19% and 5% compared with the existing methods. However, it ignores the dependency between requirements and fails to reduce both the costs and duration of a project.

(Rottoli & Casanova., 2022) proposed a method for prioritizing requirements by using fuzzy linguistic labels. The model collects various experts' opinions based on multiple decision criteria provided. Then these opinions are combined using the fuzzy aggregation operator MLIOWA considering several weights for each expert. better results are achieved in the requirement prioritization process, when assessed on three different aspects: complexity, degree of reusability and importance to costumer. However, it suffers from poor collision handling and also ignores dependency between requirements. further the absent of supporting tool is also recognized.

(Hassan et al., 2022) proposed a genetic algorithm based fuzzy TOPSIS method for requirements prioritization. The genetic algorithm used for generating the fuzzy numbers automatically using genetic algorithm. Then the linguistic variables are used by the experts through the evaluation of requirements based on different criteria. The applicability of the method is deliberated by the requirements of an institute examination system. However, it ignores the dependency between requirements, and limited to small set requirements.

(Devadas & Cholli., 2022) presented a new prioritization method called, Deep Neural Lagrange Multipler-based Multi-Aspect Large Scale Software Requirement Prioritization (DLM-MLSRP). The ultimate objective is to improve the prioritization process established on the combination of the related aspects of benefit and cost that define the requirements priority. (DLM-MLSRP) method involve four layers, specifically single input layer, two hidden layers and single output layer. The input layer contains the requirement specification gotten from the customer. The primary hidden layer implements requirement selection through Criteria Hypothesis formulation. The another hidden layer performs Pair-wise assessment using Lagrange Multipler Eigen-based function. Lastly, the output layer procedures the requirement prioritization matrix. The results show that the proposed method

DOI: https://doi.org/10.70274/jaict.2024.1.1.18.
p- ISSN 34562-3478 e- ISSN 5443-1243

outperformed SRP Tackle and IFS respectively. The time effectiveness of DLM-MLSRP method was found to be 24% and 36% better than that of two mentioned methods. The accuracy of DLM-MLSRP demonstrations 98.33% accuracy when compared with 96.6% and 93.33% of the two methods, Moreover, regarding the sensitivity of DLM-MLSRP displays 0.88 compared to 0.85 and 0.81 of the two methods. Generally, the results of DLM-MLSRP method highlight improvement in term of specificity by 8% and 20% compared to the two methods. However, it has limitations regarding the subjective use of ordinal scales and ignoring the dependency between requirements.

(ul Hassan et al., 2022) introduced a novel automated a risk-based requirements prioritization model of contract requirements, which is vital to the advances of requirements information management technologies for construction projects. In initial step the requirements associated with their risk assessment data were captured. Then utilizes a fuzzy FMEA system to compute and label the requirements with risk prioritization classes. Adopting Convolutional Neural Networks (CNN) to train a risk-based RP model, the input is the requirement text and the output is risk category (three risk aspects: severity, detectability, and probability). They used word2vec for converting requirements text to numerical data. The model was evaluated through data elicited from historical design-build project contracts. The evaluation results indicate the notable precision, recall, and f-score of 82.72%, 87.38%, and 83.97%, respectively. However, it neglects the dependency between requirements.

(Ko et al., 2024) presented a novel project requirements prioritizing method founded on the influence levels of adjusted work items. they analyze the impacts of the modification of work items on the whole project performance during the construction phase, by controlling historical change orders. Assessing the impact level make it promising to understand the negative effect of incomplete or unsatisfied requirements on cost estimates and schedules. They adopted NLP to automate project requirements classification, as it ability in comprehending the contents of project proposal documents through training. The model validated through a case study by investigating documents from resurfacing projects authenticating the feasibility and efficiency of the proposed method. The results show that It will also offer a foundation for a smarter review and considerate of project documentation and enhance decision-making for project planning. However, it ignores the dependency between requirements.

2.2 Dependency- aware AI and machine learning requirements prioritization techniques

(Alrashoud & Abhari., 2015) presented a mathematical interpretation to model that tactile the uncertainty open issues in human estimation and their limited knowledge. The model adopted Fuzzy Inference System (FIS). The proposed model involves three processes: I) Preprocessing, in which the weighted importance of the requirements is calculated, the dependency relationship is represented, and the FIS engine is built; II) Ranking process in which the FIS engine is adopted to rank for each requirement. The inputs to the FIS engine are: weighted importance, final effort regarding dependency constraints, and the risk; III) Plan-generation process, in which the highest ranked requirements are allocated to the release plan. The applicability of proposed model is presented using a set of twenty requirements provided by four stakeholders. The findings highlighted that the FIS model has achieved a higher degree of satisfaction when compared to a genetic algorithm-based model. However, it handles small set of requirements and only covers two types of dependency (combination and implication).

(Allex et al., 2016) proposed Interactive Next Release Problem (iNRP) model for requirement prioritization. The model adopts Least Median Square (LMS) and Multilayer Perceptron (MLP) techniques. The iNRP model consist of composed of three different modules with separate responsibilities: interactive genetic algorithm, interactive module, and learning model. In the iNRP model two architectural settings is specified by decision maker(DM), the weight of the tacit preferences is compared to the explicit ones for the suitability calculation. Then, the learning process is executed using the set of samples captured in the preceding stage as a training dataset. However, the performance of this learning model has not been evaluated.

(Shao et al., 2017) proposed a semi-automatic requirements prioritization approach, named Drank. Drank takes the dependencies among requirements and the stakeholders’ preferences into consideration. Rank utilizes RankBoost algorithm to produce requirement prioritization formula in

DOI: https://doi.org/10.70274/jaict.2024.1.1.18.
p- ISSN 34562-3478 e- ISSN 5443-1243

subjective manner, produce requirement dependency graphs (RDGs) based on the contribution dependencies and business dependencies among the requirements, next analyze the contribution order to compute the contribution of each requirement by adopting PageRank algorithm to finally assimilate the final requirements prioritization. a controlled experiment made to validate the DRank efficiency, constructed on comparisons with Case Based Ranking, AHP, and EVOLVE. The findings establish that DRank is minimizes time-consuming and more effective as compared with alternative approaches. A simulation experiment proves that concerning the dependencies among the requirements can increase the accuracy of the final prioritization sequence. However, this work seems to provide only the contribution and business dependencies. Further, the authors highlight that their approach is still motivated to the issue of subjectivity especially in the process of requirements evaluation.

(Gupta & Gupta., 20١٨) proposed a semi-automated dependency- based collaborative requirement prioritization approach named (CDBR), which uses an execute-before-after (EBA) connection among requirements, linguistic values, and a machine learning algorithm to reduce variances in opinion among stakeholder and developer and improve final priority estimation. The CDBR focused on three major problems which are usually overlooked in an existing research: dependencies between requirements, stakeholder and developer's collaboration and scalability. To get final agreeable implementation priorities, CDBR uses the Particle Swarm Optimization (PSO) algorithm to reduce disputes among stakeholders and developers' ranking. To evaluate the approach's performance tow scenario are chosen: in the first one, nine different requirement sets are applied to evaluate the suggested approach's performance in terms of managing scalability. stakeholders and developer's priority for all these requirements is determined randomly. The developer's priority is computed using a dependency matrix that is similarly created randomly while keeping the density of the matrix in mind. The higher the density, the further dependencies there are in the system, and hence the more complex it is. In the second scenario, validation on the case study of cargo booking management in a warehouse (CBMW) is chosen, the CDBR, interactive genetic algorithm (IGA), and AHP are used. The precision of the results is identified by comparing the CDBR against AHP and IGA priority lists using the Analysis of Variance (ANOVA) test method. The outcomes are accurate and equivalent in terms of scalability, accuracy, and stakeholder and developer variances levels. In terms of efficiency and processing time, CDBR beats AHP and IGA. Despite an improvements detected in prioritization results, and processing time together. However, it ignores the errors (false positive rate) implicated during the prioritization process.

(Gupta & Gupta., 20١٨) proposed a dependency collaborative requirements prioritization method, where both stakeholder and developers are involved in providing ranking to requirements. the initial priority on the base of three criteria's namely, urgency, necessity and importance to business value. Additionally, a dependency classification method and weight assignment method to support developers in making right decision during prioritization. Good result achieved in term of low complexity when compared with existing method. However, it limited to three type of dependency, suffer from biases issues, lack of automation.

(Misaghian et al., 2019) presented a requirement prioritization approach, using fuzzy graph algebra weighted page rank algorithm tensor decomposition. In this approach, the requirements order provided by tensor decomposition mixed with the dependency order. Good result achieved in terms of improvements in accurate, time-consuming, and ease of use. However, only the increase/decrease cost dependency type is supported in this work, and it ignores the stakeholders' conflicts.

(Inayat et al., 2019) used a modified PageRank algorithm to prioritize the specified requirements. Good result obtained in term of efficiency and accuracy as compared with five existing requirements prioritization methods. However, the dependency between requirements is limited to dependency type defined in Ecore meta-model and the lack of automation is also seen.

(Gupta & Gupta., 2022) proposed a scalable framework for prioritizing the requirements of obtaining the inputs from both stakeholders and developers. The stakeholders offer their inputs according to the project criteria Intuitionistic Fuzzy Approach (IFS) is utilizes to support stakeholder's opinion. on the other hand, the developers, the developers, on the other hand, offer dependency graph as their part of the input. Next, Weighted Page rank adopted this dependency graph to calculate the initial priority

DOI: https://doi.org/10.70274/jaict.2024.1.1.18.
p- ISSN 34562-3478 e- ISSN 5443-1243

values. The jointed priority values from both stakeholders and developers are finally used to calculate the final priority. validation result indicates that this framework is capable of providing accurate and comparable results by handling technical constraints of dependency as compared to existing techniques. However, the requirement prioritization accuracy was not improved with less time, and its limited to one type of dependency (technical constraints) and the absent of automation is also noticed.
(Devadas & Cholli., 2022) proposed a novel method called the Interdependency-aware Qubit and BrownRoost Rank (IQ-BR) method to prioritize the massive set of requirements. IQ-BR begin with the identified FR and NFR that are required to be prioritized. Then calculate the dependency among them using dependency value matrix (DVM). Finally, instable and interdependent requirements are selected using Quantum Optimization functions. And prioritizing the requirements adopting the BrownBoost Learning model that supports in precise decision-making for ranking a set of optimal requirements. A fairly good result had been achieved in precisely prioritizing requirements and reducing the noise in a large set requirement prioritization. The simulations conducted using IQ-BR, CDBR, and IFS highlighted that the accuracy percentage was detected to be 95%, 90%, and 91.66%. However, it fails to address uncertainty and test suite execution issue among diverse stakeholders for large scale software requirement prioritization and the absent of supporting tool is also seen. Moreover, (IQ-BR) not provide a mechanism to handle parallel prioritization which could accelerate the software project delivery.
(Eldrandaly., 2023) presented a multi-criteria decision making (MCDM) framework for requirements prioritization, adopting the DEMATEL and TOPSIS methods in the neutrosophic environment. using the type-2 neutrosophic numbers (T2NNs) based DEMATEL method to compute and rank the criteria importance. The DEMATEL method used in the framework handles the interdependency between the requirements. Then the T2NN-based TOPSIS is used to rank the requirements. Lastly, the proposed framework applicability is verified with the help of a numeric case study. Its proven thatneutrosophic approach adopted in this study addresses the fuzziness and vagueness in the stakeholders' decisions, making it potential for stakeholders to use linguistic terms as an alternative of numbers and scales which can be understood variously by everyone, which can lead to inaccurate results.
However, it needs to be tested on a large project for further validation.

3. Discussion

There are numerous machine learning techniques proposed for automatic determination of requirements prioritization. Several aspects influence RP, however, the challenging aspect in RP is handling requirements dependency (RD), which means the requirements are dependent or reliant on each other. Incorrect handling of requirements dependencies could lead to software development failures (Noviyanto et al., 2023). Moreover, prioritizing these requirements without considering their dependencies can negatively affect the accuracy of the final prioritization results. Nonetheless, this attribute is rarely considered by the authors of requirement prioritization studies (Noviyanto et al., 2023). Many techniques for handling requirements dependencies mentioned above, and we can classify some prioritization techniques considering the dependency between requirements using Machine Learning Techniques as follow:

1. Tensor and Fuzzy Graphs
2. Matrix Drank((PageRank, RankBoost)
3. Meta-model and PageRank algorithm.
4. Dependency Graph and Weighted page rank algorithm
5. Collaborative requirement prioritization approach (CDBR).
6. DVM and BrownBoost Rank Requirement Prioritization Learning model.
7. Collaborative requirement prioritization Method.
8. Supervised Classification Techniques.
9.Interactive Next Release Problem (iNRP)
10. integrating Active Learning with Ontology Based Retrieval
11. Fuzzy Inference System

DOI: https://doi.org/10.70274/jaict.2024.1.1.18.
p- ISSN 34562-3478 e- ISSN 5443-1243

We notice that Although there is a trend in adopting ML techniques, many methods are built in different techniques, and some of them have approved their evaluations with many requirements no greater than 50 (Small or Medium Scale) i.e. lack of scalability. We can see also that few of these studies seek to handle the dependencies between requirements; and lack of supporting tool is also observed. Table 1 show some dependency-aware requirements prioritization techniques based AI and ML.
It explains the comparison between the Method (Algorithm Type), Prioritization Criteria, technique used, Issues Addressed, and Evaluated Data Level, which is mention in Table 1.

Table 1: Some dependency-aware requirements prioritization techniques based ML

Study ID	Method (Algorithm Type)	Prioritization Criteria	Techniques	Issues Addressed	Evaluated data Level	Limitations
1	Tensor	Stakeholders ranking	Fuzzy graph algebra weighted page rank algorithm tensor decomposition	Consider FR, NFR stakeholders ranking scalability	Small scale	Only (increase/decrease cost) covered and the absent of automation tool
2	Combined Machine learning and link analysis technology	Stakeholders Ranking Dependencies Effectiveness comparison	Tree for ranking RankBoost Weighted PageRank	Dependencies risk factors cost benefits	Medium scale	Limited to dependencies defined in in i* meta -model and the absent of supporting tool
3	Machine-learned ranking	Stakeholders ranking dependencies risk factors cost benefits	PageRank algorithm	Dependencies consider NFR reduce Manual effort	Medium scale	Limited to dependency type defined in Ecore meta-model Lack of automation
4	Machine-learned ranking	Stakeholders ranking developer ranking	Dependency graph Weighted page rank algorithm	Communication among stakeholders scalability dependency	Large scale	its limited to (technical constraints) dependency the absent of automation
5	Optimization algorithm.	Stakeholders Ranking Developer ranking Importance Urgency	Graphs Dependency matrix Particle Swarm Optimization (PSO)	Dependencies Communication among stakeholder and	Medium scale	A linguistic value is less accurate than the numerical value lack of automation

DOI: https://doi.org/10.70274/jaict.2024.1.1.18.
p- ISSN 34562-3478 e- ISSN 5443-1243

				developers and Scalability.		
6	Optimization algorithms	Customers ranking	BrownBoost Rank, and Quantum Optimization	both FRs and NFRs considered address volatile and dependency performance(time and accuracy)	Large scale	Ignore the different points of views between stakeholders
7	Classification and weight assignment method	Stakeholders Ranking Developer ranking	Weighted score Critical path method (CPM)	Communication among stakeholder Urgency Necessity and importance	Small scale	Limited to three type of dependency. Suffer from biases issues Lack of automation
8	Machine learning	Dependencies engineering expert opinions	Supervised classification techniques based on text mining	Automation Dependency precision	Medium	It handles one type of dependency (requires).
9	Machine learning	decision maker tacit assessments interdependencies stakeholders ranking importance budge	Interactive Next Release Problem (iNRP) and interactive genetic algorithm	Interdependencies Release plan	Small Medium and large	The performance of this learning model has not been evaluated.
10	Machine learning	domain expert ranking Dependency	Active Learning with OntologyBased Retrieval	research questions are investigated on the two	Median and large	It limited to three types of dependencies. The ontology depends on context engineering

DOI: https://doi.org/10.70274/jaict.2024.1.1.18.
p- ISSN 34562-3478 e- ISSN 5443-1243

				industrial data sets Dependency Accuracy automation		
11	Fuzzy logic	Stakeholder ranking. importance, risk, required effort and Dependency	Fuzzy Inference System (FIS)	Dependency Risk	Small	It handles small set of requirements and only covers two types of dependency (combination and implication).

The effectiveness of any prioritization method denotes by size to produce a fast and accurate prioritization. Thus, it is an important directory for evaluating the quality of a prioritization method. Table 2 shows the accuracy of the dependency-aware requirements prioritization techniques.

It explains the comparison between the accuracy, technique used, and type of the data set of eleven techniques to prioritize requirements, which is mention in Table 1.

Table 2: The accuracy and type of data se of the dependency-aware requirements prioritization techniques

No.	Algorithm	Technique	Type of dataset	Accuracy
1	Tensor and Fuzzy Graphs	weighted page rank algorithm based on the fuzzy concept	a distance learning management system (DLMS) project.	24%,
2	Combined Machine learning and link analysis technology (Drank)	RankBoost Weighted PageRank	Book Trading System (BTS) and Library Management System (LMS)) in Software Company located in Wuhan, China.	Not stated
3	Machine learning	PageRank Algorithm	a smart home system.	Not stated
4	Fuzzy Logic	Intuitionistic Fuzzy Approach (IFS)	Software Requirement Risk Prediction Dataset. https://zenodo.org/record/1209601#.XJsL_JgzaM-	Not stated.

DOI: https://doi.org/10.70274/jaict.2024.1.1.18.
p- ISSN 34562-3478 e- ISSN 5443-1243

		Weighted page rank algorithm		
5	Optimization algorithm.	Particle Swarm Optimization (PSO)	cargo booking management in a warehouse (CBMW).	Not stated
6	Optimization algorithm.	, BrownBoost Rank, and Quantum Optimization	, "Software requirements dataset," https://www.kaggle.com/ iamsouvik/software-requirements-dataset.	95%
7	Decision Making	Collaborative requirement prioritization method a weighted score method is used	Illustration example of 14 requirements	Not stated
8	Machine learning	Supervised Classification Techniques	a real-world dataset requirements for a sports watch. And Graz University of Technology (http://www.tugraz.at).	82%
9	Machine learning	Interactive Next Release Problem (iNRP)	two experiments were performed in the empirical study: (a) artificial experiment: a simulator was employed. and (b) participant-based experiment: a group of software engineer practitioners was invited to solve a real-world instance using the proposed approach.	has not been evaluated .
10	Machine learning	Integrating Active Learning with OntologyBased Retrieval	two industrial datasets, namely Siemens Austria and Blackline Safety Corp Canada.	86%
11	Fuzzy logic	Fuzzy Inference System (FIS)	A set of 20 requirements simulated data	Not assessed

Most of prioritization method check their accuracy by comparing it to existing one, for instance (Misaghian et al., 2019), compared their methods to AHP, TOPSIS, and EVOLVE. AHP considered as the standard approach in several domains and studies (Misaghian et al., 2019), (Shao et al., 2017). The accuracy of the results in their proposed method is greater than that in the AHP and TOPSIS, which offer prioritized requirements by focusing on stakeholders' preferences and human judgment with careless of the requirement dependencies, and EVOLVE which detects business dependencies among requirements. The AHP, TOPSIS, and CBRank which, emphasis on prioritization based on stakeholder

DOI: https://doi.org/10.70274/jaict.2024.1.1.18.
p- ISSN 34562-3478 e- ISSN 5443-1243

preferences, and do not reflect the dependencies among the requirements. Therefore, their prioritization can only reflect the preferences of the stakeholders (Shao et al., 2017).

Many techniques can be improved by joining the strength of two or more strong algorithms to deal with special cases that didn't follow in developing effective requirement prioritization techniques which are improper prioritized by some weak or single techniques.

4. Conclusion

The increased request of complex software systems by stakeholders with massive set of requirements in the last years, raised up the need for effective requirements prioritization techniques. an effective requirements prioritization must be considered the dependency among requirements order to avoid system failure.

In this paper, we had presented and discussed numerous papers and articles work on prioritizing software requirements, machine learning techniques, and dependency computation. and explain the strength and weakness points in them.

The benefits of related works serve many purposes, some of which relate directly to reviewing, the person choose the submission will use the referenced papers to find good evaluators, evaluators will look at the references to decide that the submission cites the suitable work, everybody will use the section to recognize the paper's contributions provided the state of existing research and future researchers will look to the Related Work unit to detect other papers they should read.

5. Acknowledgment

This work was supported by the Deanship of Scientific Research at Omdurman Islamic University, Omdurman, Sudan.

Funding:

"This research received no external funding"

References

Noviyanto, F., Razali, R., & Ahmad Nazri, M. Z. (2023). Understanding requirements dependency in requirements prioritization: a systematic literature review. International Journal of Advances in Intelligent informatics, 9(2).

Duan, C., Laurent, P., Cleland-Huang, J., & Kwiatkowski, C. (2009). Towards automated requirements prioritization and triage. Requirements engineering, 14, 73-89.

A Perini, A., Susi, A., & Avesani, P. (2012). A machine learning approach to software requirements prioritization. IEEE Transactions on Software Engineering, 39(4), 445-461.

Achimugu, P., & Selamat, A. (2015). A hybridized approach for prioritizing software requirements based on K-Means and evolutionary algorithms, Computational Intelligence Applications in Modeling and Control, vol. 575, A. T. Azar and S. Vaidyanathan, Eds. Cham, Switzerland: Springer, 2015, pp. 73–93, doi: 10.1007/978-3-319-11017-24.

Babar, M. I., Ghazali, M., Jawawi, D. N., Shamsuddin, S. M., & Ibrahim, N. (2015). PHandler: an expert system for a scalable software requirements prioritization process. Knowledge-Based Systems, 84, 179-202.

Gupta, A., & Gupta, C. (2018). A collaborative approach for improvisation and refinement of requirement prioritization process. Journal of Information Technology Research (JITR), 11(2), 128-149.

Singh, Y. V., Kumar, B., Chand, S., & Kumar, J. (2018). A comparative analysis and proposing 'ANN fuzzy AHP model 'for requirements prioritization. Int. J. Inf. Technol. Comput. Sci, 10(4), 55-65.

Sandanasamy, A., & Selvi, R. T. (2019). A Quality Based Software Requirement Prioritization Using Takagi-Sugeno Neuro Fuzzy Inference.

Singh, Y. V., Kumar, B., Chand, S., & Sharma, D. (2019). A hybrid approach for requirements prioritization using logarithmic fuzzy trapezoidal approach (LFTA) and artificial neural network (ANN). In Futuristic Trends in Network and Communication Technologies: First International Conference, FTNCT 2018, Solan, India, February 9–10, 2018, Revised Selected Papers 1 (pp. 350-364). Springer Singapore.

Sher, F., Jawawi, D. N., Mohammad, R., Babar, M. I., Kazmi, R., & Shah, M. A. (2020). Multi-aspects Intelligent Requirements Prioritization Technique for Value Based Software Systems. In Intelligent Technologies and Applications: Second International Conference, INTAP 2019.

Ali, S., Hafeez, Y., Asghar, S., Nawaz, A., & Saeed, S. (2020). Aspect-based requirements mining technique to improve prioritisation process: multi-stakeholder perspective. IET Software, 14(5), 482-492.

Hujainah, F., Bakar, R. B. A., Nasser, A. B.Al-haimi, B., & Zamli, K. Z. (2021). SRPTackle: A semi-automated requirements prioritization technique for scalable requirements of software system projects. Information and Software Technology, 131, 106501.

Gambo, I. P., Ikono, R., Iroju, O. G., Omodunbi, T. O., & Zohoun, O. K. (2021). Hybridized ranking model for prioritizing functional software requirements: Case study approach. International Journal of Software Innovation (IJSI), 9(4), 19-49.

Sadiq, M., & Devi, V. S. (2021). A rough-set based approach for the prioritization of software requirements. International Journal of Information Technology, 1-11.

Ahmad, S., Rizawanti, R., Woodings, T., & Jalil, I. E. A. (2022). MCBRank Method to Improve Software Requirements Prioritization. International Journal of Advanced Computer Science and Applications, 13(7).

Chua, F. F., Lim, T. Y., Tajuddin, B., & Yanuarifiani, A. P. (2022). Incorporating semi-automated approach for effective software requirements prioritization: A framework design. Journal of Informatics and Web Engineering, 1(1), 1-15.

Devadas, R., & Cholli, N. G. (2022). PUGH Decision Trapezoidal Fuzzy and Gradient Reinforce Deep Learning for Large Scale Requirement Prioritization. Indian Journal of Science and Technology, 15(12), 542-553.

Rottoli, G. D., & Casanova, C. (2022). Multi-criteria and multi-expert requirement prioritization using fuzzy linguistic labels. ParadigmPlus, 3(1), 1-18.

Nazim, M., Hassan, T., Mohammad, C. W., & Sadiq, M. (2022) An Integrated Method for the Selection of Software Requirements Using Fuzzy TOPSIS and Genetic Algorithm. International Journal of Innovative Science and Research Technology.

Devadas, R., & Cholli, N. G. (2022, March). Multi aspects based requirements prioritization for large scale software using deep neural lagrange multiplier. In 2022 International Conference on Smart Technologies and Systems for Next Generation Computing (ICSTSN) (pp. 1-6). IEEE.

ul Hassan, F., Le, T., Le, C., & Shrestha, K. J. (2022, June). Automated prioritization of construction project requirements using machine learning and fuzzy logic system.

In Proc. 9th International Conference on Construction Engineering and Project Management (ICCEPM) (pp. 20-23).

Ko, T., Lee, J., & David Jeong, H. (2024). Project Requirements Prioritization through NLP-Driven Classification and Adjusted Work Items Analysis. Journal of Construction Engineering and Management, 150(3), 04023171.

Alrashoud, M. and Abhari, A. (2015) Intelligent Automation & Soft Computing Perception-Based Software Release Planning, Intell. Autom. Soft Comput., vol. 21, no. 2, pp. 175–195, doi: 10.1080/10798587.2014.960229.

Allex, A., Matheus, A., I. Yeltsin, Dantas, A., and J. Souza, (2016). An Architecture based on interactive optimization and machine learning applied to the next release problem, Autom. Softw. Eng., vol. 24, no. 3, pp. 623–671, doi: 10.1007/s10515-016-0200-3.

Shao, F., Peng, R., Lai, H., & Wang, B. (2017). DRank: A semi-automated requirements prioritization method based on preferences and dependencies. Journal of Systems and Software, 126,141-156.

Gupta, A., & Gupta, C. (2018). CDBR: A semi-automated collaborative execute-before-after dependency-based requirement prioritization approach. Journal of King Saud University-Computer and Information Sciences, 34(2), 421-432.

Gupta, A., & Gupta, C. (2018, August). Towards dependency based collaborative method for requirement prioritization. In 2018 Eleventh International Conference on Contemporary Computing (IC3) (pp. 1-3). IEEE.

Misaghian, N., Motameni, H., & Rabbani, M. (2019). Prioritizing interdependent software requirements using tensor and fuzzy graphs. Turkish Journal of Electrical Engineering and Computer Sciences, 27(4), 2697-2717.

Abbas, M., Inayat, I., Jan, N., Saadatmand, M., Enoiu, E. P., & Sundmark, D. (2019, December). Mbrp: Model-based requirements prioritization using pagerank algorithm. In 2019 26th Asia-Pacific Software Engineering Conference (APSEC) (pp. 31-38). IEEE.

Gupta, A., & Gupta, C. (2022). A novel collaborative requirement prioritization approach to handle priority vagueness and interrelationships. Journal of King Saud University-Computer and Information Sciences,34(5)2288 -2297.

Devadas, R., & Cholli, N. G. (2022). Interdependency Aware Qubit and Brownboost Rank Requirement Learning for Large Scale Software Requirement Prioritization. International Journal of Computing and Digital Systems, 11(1), 625-635.

Eldrandaly, B. K. (2023). A Framework of Type-2 Neutrosophic for Requirements Prioritization. Neutrosophic Sets and Systems, 53, 421-432.

DOI: https://doi.org/10.70274/jaict.2024.1.1.18.
p- ISSN 34562-3478 e- ISSN 5443-1243