Marketing and Branding Research

WWW.CIKD.CA

An Empirical Evaluation of Requirements Prioritization Techniques

Naila Jan[1*], Irum Inayat[2], Muhammad Abbas[3]

[1, 2]SERL, National University of Computer and Emerging Sciences, Islamabad, Pakistan

[3]RISE SICS, Research Institutes of Sweden, Västerås, Sweden

ABSTRACT

In today's time and budget intensive software development market, quick delivery is the basic motive of teams. Software development teams strive to gain customer satisfaction by all possible means. Requirements prioritization is the most challenging customer input dependent task in the software development life cycle that decides the fate of a project. Selection of a well-suited requirements prioritization technique may result in customer satisfaction and on-time delivery time. Literature reports on many requirements prioritization techniques in practice. However, each has its own features that can outperform the rest for a certain case. Therefore, this research is conducted to empirically evaluate the existing techniques in terms of certain quality measures (i.e., accuracy, efficiency, and scalability). The selected techniques are evaluated for the small, medium and large scale of requirements sets. For that, we selected five existing techniques that are multi-criteria-decision-making techniques and have user involvement (i.e., Analytical Hieratical Process (AHP), Analytical Network Process (ANP), FuzzyAHP, FuzzyANP and Interactive Genetic Algorithm (IGA)). The experimental results showed that among the five selected techniques, FuzzyAHP is the most efficient and accurate technique for the large dataset of requirements.

Keywords:
Requirements Prioritization, Prioritization Techniques, AI-based Prioritization Techniques, Manual Prioritization Techniques, Empirical Evaluation

Received
30 September 2020
Received in revised form
20 October 2020
Accepted
26 October 2020

***Correspondence:**
nailaneena@gmail.com

Requirements prioritization phase identifies which requirements will go first in a certain release (Pergher, Massimilianopergherstud-infunibzit, & Rossi, 2013). This helps in identifying risk factors early in the project development lifecycle and aids to minimize them. Requirements prioritization is the phase of software development that involves measurements, reasoning, and computation (Achimugu, Selamat, Ibrahim, & Naz, 2014). Requirement prioritization is also used for cost estimation of the software system that helps to achieve the business value through high quality and minimum maintenance effort (Vaidya & Kumar, 2006).

Two kinds of factors influence the overall process of requirement prioritization internal factors (i.e., related to the project) and external factors (i.e., stakeholders related and prioritization related). The example of the internal factors is the risk of the requirements, its development cost and the dependency of the requirements while the external factors are the stakeholder’s preference (i.e., the candidate requirements specified by the stakeholder's group) and the business goals that are the prioritized requirements meet the market value, etc. (Vaidya & Kumar, 2006).

The complexity of the requirement prioritization process increases in terms of time and effort with an increase in the number of requirements (Agrawal, 2016). For instance, AHP can handle up to 20 requirements because requirements are compared based on possible pairs using n(n-1)/2 comparisons, due to this comparison it requires four times the effort or time by just doubling the number of requirements in a list (Ahuja & Batra, 2018; Jawale & Bhole, 2017) Thus scalability emerges as one of the limitations in requirements prioritization along with accuracy (Agrawal, 2016). Since software varies in size and with constant evolution number of requirements is prone to increase. Therefore, the requirements prioritization techniques should be able to handle large data (i.e., requirements) efficiently. Use of artificially intelligent techniques (Daneva, 2013; Herrmann & Paech, 2009; Tonella, Palma, Tonella, & Susi, 2010) and meta-heuristic mechanism (Mustafa & Zainuddin, 2014) solves the problems of scalability and accuracy to some extent (Daneva, 2013; Herrmann & Paech, 2009).

In this research, we aim to empirically evaluate the aforementioned AI-based RP techniques; i.e., Analytical hieratical Process (AHP) (Nidhra, Satish, & Ethiraj, 2012), Analytical Network Process (ANP) (Dabbagh et al., 2016), FuzzyAHP (Perini et al., 2007), FuzzyANP (Mczara et al., 2015) and (IGA) Interactive Genetic Algorithm (Tonella, Palma, Tonella, & Susi, 2010) to analyze (i) the accuracy and (ii) the efficiency of the said techniques according to the variable number of requirements. We measure efficiency as the total time of task completion (recording the starting time and the ending time) and accuracy as the percentage of its similarity of the results with the originally prioritized list.

Background

There are a lot of requirements prioritization techniques for both manual and automated prioritization. AI-based prioritization techniques are also proposed using different algorithms that are fuzzy logic and evolutionary algorithms (Agrawal, Singh, & Sharma, 2016; Jawale, Patnaik, & Bhole, 2017; Tonella et al., 2010). But a very limited focus on the empirical validation of these techniques. And the empirically evaluated techniques still have some limitation that is a lack of focus on scalability, easy to use and learnability (Pergher et al., 2013).

There are still some studies that do not fully address the scalability, easy to use and learnability (Pergher et al., 2013), especially the recent AI proposed Algorithm for requirements prioritization (Ahuja & Batra, 2018; Jawale et al., 2017; Tonella et al., 2010).

Results of a recently conducted systematic literature review of requirements prioritization techniques show AHP as the most cited and accurate technique with limitations like computational complexity (Achimugu et al., 2014). Some other techniques like ANP, Bubble sorting, CBRanking, MosCow, cumulative voting, and ranking also have limitations like scalability and rank update (updating, deleting or adding a requirement). However, it is also reported that most of the AI-based RP techniques are not empirically evaluated for variable size of requirements and for multiple prioritization factors.

Traditional Techniques

Many researchers have been conducted on manual requirements prioritization techniques. Different manual techniques were proposed and evaluated for different measures. For example, Herrmann and Paech (2009) conducted an empirical experiment for multi-level hierarchical requirements. The study uses a hierarchical cumulative voting (HCV) technique to prioritize requirements that depend on the number of elements in each prioritization block. The goal was to identify different conditions that how they can compensate for the priority of requirements in the given hierarchy. The compensation factor was identified by a priority calculator for 27 functional requirements. The post-test shows that the subjects were happy with HCV as a prioritization technique for hieratical requirements. The study is not generic for all hierarchical projects; their priority may be different for different scenarios. Daneva et al. (2013) studied ranking and risk-based-prioritization of requirements to identify the accuracy and ease-of-use. The authors used a real prioritized list of requirements to be compared with the results of the aforementioned techniques, statistically. They focused only on risky requirements and the results showed that ranking results were close enough to the originally prioritized list. However, the results cannot be generalized because the dataset comprised of only ten requirements.

Mustafa and Zainuddin (2014) conducted a theoretical study to analyze nine prioritization techniques from the literature and to identify the limitations of those techniques. The results showed that the techniques did not focus on the human behavior factor for prioritization. The author proposed a hybrid technique which is the combination of AHP, Numerical assignment, Ranking, and MoSCoW. The proposed technique was statistically tested which showed a high ratio for ease-of-use and customer satisfaction. The study was empirically evaluated for real projects; however, the size of the dataset was not mentioned.

An empirical study (Nidhra, Satish, & Ethiraj, 2012) was conducted for AHP and Numerical Assignment techniques to identify the complexity and time of execution. A hybrid technique was proposed called NAcAHP and was evaluated on a large data-set (40 requirements). The results showed that the hybrid technique was more efficient and scalable than AHP and Numerical Assignment. However, the s experiment was performed on university students so the results might not be generalized for industrial projects.

Dabbagh, Peck, Reza, and Parizi (2016) conducted a control experiment to identify the accuracy, ease-of-use, and efficiency of AHP, IPA, and HAM-based approaches. All the techniques were used for prioritization of twenty requirements (both functional and non-functional). The statistical tests showed that IPA outperformed AHP and HAM. However, the major limitation was the size of the dataset.

AHP and CBRanking were empirically evaluated to identify a more accurate technique for requirements prioritization. The dataset used comprised of twenty requirements of a real-world project and the subjects include Ph.D. researchers and industrial experts. The statistical results showed that AHP outperformed CBRanking in terms of accuracy. The major limitation is a small data set and missing factors like scalability and efficiency (Perini, Ricca, Bruno, Irst, & Susi, 2007).

The literature review of empirical studies of manual requirements prioritization techniques shows that the largest dataset used till date comprises not more than forty requirements (Ahuja & Batra, 2018). Therefore, in our study, we aim to consider a larger dataset to focus on the scalability issue of the selected techniques in relation to the accuracy and efficiency of the techniques.

AI-Based Techniques

Some recent work is also carried out in AI-based requirements prioritization techniques. The proposed techniques are mostly based on Fuzzy logic and evolutionary algorithms. For example, Mczara, Sarkani, Holzer, and Eveleigh (2015) proposed a semi-automated method was proposed for RP called SMT- and NLP-based interactive requirements prioritization (SNIPR). It is a linguistic tool based on Natural Language Processing (NLP) and satisfiability modulo theories solvers (SMT solver). SMT solver is a common technique that translates precondition, post-condition, loop condition and assertions into SMT formula to determine if all properties can hold. This method was evaluated through a control experiment for 20 requirements to measure the accuracy, ease of use and performance as compared to the weighted sum method (WSM). The results showed that SNIPR was significantly better than WSM.

An evolutionary algorithm was used to prioritize requirements called Interactive Genetic Algorithm (IGA). The proposed algorithm was evaluated on a real case study comprising forty-nine functional requirements. The algorithm was statistically tested to determine the robustness and effectiveness of the approach (Tonella et al., 2010). The proposed technique IGA was significantly better but this approach should be compared with other AI prioritization techniques to validate the results.

Another GA based technique was proposed to prioritize requirements called Least-Square-Based Random Genetic Algorithm (LSRGA) and was empirically evaluated to measure the performance and compared with Tonella’s approach that is IGA (Ahuja & Batra, 2018). The proposed technique was evaluated for 5 functional requirements and the statistical test shows that LSRGA’s performed better than IGA (Mczara et al., 2015). The proposed method could be tested for other factors like accuracy, consistency, and performance for a larger set of requirements to validate the results.

A method called Agile based Risk Rank (AR-Rank) was proposed to prioritize requirements to identify and prioritize risk factors in agile methodology. The proposed method uses Particle Swarm Optimization (PSO) for each iteration to optimize risk factors. The method was evaluated by using five requirements and the results showed that the proposed method outperformed AHP (Agrawal et al., 2016). However, the method was not empirically compared with other AI-based prioritization techniques.

An Adaptive Fuzzy Hierarchical Cumulative Voting (AFHCV) technique was proposed for runtime prioritizing requirements. The proposed technique adds the adaptive mechanism to the existing Fuzzy hierarchical Cumulative voting (FHCV), the adaptive feature helps to add a requirement and to alter its ranking at runtime. The proposed technique was empirically evaluated by comparing it with FHCV to identify the real priority of the requirements. The technique was evaluated by comparing its results with the existing priority and the proposed technique results were close to the actual results (Jawale et al., 2017). Table 1 Summarizes the literature review on existing requirements prioritization methods.

Table 1 shows that almost all of the studies were evaluated for a limited number of requirements (not more than 50). The AI-based prioritization techniques also ignored the usage of large datasets and empirical evaluation of the proposed work or did on a specific case only (Ahuja & Batra, 2018; Tonella, Susi, & Palma, 2013). To close this gap, in this study we aim to focus on the scalability issue. In this research an empirical evaluation for the AI-based requirements prioritization techniques (i.e., AHP, ANP, FAHP, FANP, and IGA) were conducted to measure the accuracy and efficiency of the selected techniques for different sizes of requirements dataset.

Table 1

Summary of Existing Requirements Prioritization Techniques

Technique	Ref	Evaluation Metrics	Dataset size and type	Evaluation method
AHP, ANP, ST, hierarchy AHP, spanning tree matrix, priority group, and bubble sort	(Vestola, 2010)	Performance, errors occurrence, and reliable results	12-15 functional and non-functional requirements	By the participants themselves
AHP, CBRanking	(Perini et al., 2007)	Accuracy	20 functional requirements	Questionnaires & statistical analysis
AHP-GORE-PSR	(Sadiq & Hassan, 2017)	Consistency ratio	8-10 functional and non-functional requirements	Compared the statistical results with AGORE.
AHP, Numerical Assignment	Nidhra, Satish, & Ethiraj, 2012)	Complexity, time to execution	40 functional requirements	Statistical analysis (Paired T-test)
Adaptive Fuzzy cumulative voting (AFHCV), (FHCV)	(Jawale et al., 2017)	Realistic priority	11 functional requirements	Compared with baseline
A fuzzy approach to prioritize usability requirements	(Gulzar, Sang, Ramzan, & Kashif, 2017)	Usability factors, effectiveness, efficiency	12-15 non-functional requirements	Statistical analysis
AR-Rank (using PSO)	(Agrawal et al., 2016)	Risk factors	5 non-requirements	Compared the ranking of identified risk with AHP
Hieratical commutative voting (HCV)	(Herrmann & Paech, 2009)	Accuracy	27 functional requirements	Questionnaires filled by students and statistical tests
IPA, AHP, HAM	(Dabbagh et al., 2016)	Accuracy, time-consumption and easy-to-use	20 functional and non-functional requirements	Post-experiment questionnaires and statistical analysis
Interactive Genetic Algorithm (IGA)	(Tonella et al., 2010)	Effectiveness, Robustness	49 functional requirements	Case study and statistical analysis
The ranking, risk-based prioritization	(Herrmann & Paech, 2009)	Easy to use, accuracy, realistic priorities	10 requirements	Questionnaires & statistical tests
SNIPR, WSM	(Mczara et al., 2015)	Accuracy, time, performance, learn-ability	20 functional requirements	Statistical tests

Our study contributes to identify the better performing technique for a specific size of the dataset in terms of accuracy and efficiency. We measured the accuracy by comparing the results of each technique to the baseline prioritized list. The baseline prioritized list was created by getting our dataset of 100 requirements prioritized through 30 graduate students using the factors of stakeholders priority, risk, cost, and dependency of the requirements.

Experimental Design

We experimented with three iterations for three different sizes of a dataset comprising 30, 60 and 100 requirements. The experiment was conducted to estimate the accuracy and efficiency of the selected techniques with respect to the varying size of the dataset.

Baseline Preparation

We collected the software requirements of the “Home Automated System”. The set of requirements comprised of both functional and non-functional requirements. The requirements were later prioritized based on three factors; i.e., stakeholder’s priority, risk factor, and the cost (sum of two sub-factors the development time and the requirements dependency). The requirements were ranked by thirty graduate students of MS Software Engineering registered in the course of Advanced Software Requirements Engineering. The students already covered the RP module in class and were familiar with the basic concepts. This task was performed as a class activity during regular class. The students were briefed about the home automation system through a PowerPoint presentation. Then the students were asked to rank the set of 100 requirements on the scale of 1 to 9 (from lower to higher) according to the aforementioned mentioned factors. The factors were clearly defined in the sheet and were discussed before the activity started. The ranks of all the students were evaluated by the researchers of the same institute and then the average of 28 students was used as a baseline priority for the requirements.

Requirements Prioritization Techniques Selection

As we know that requirements prioritization is a decision-making process and it involves human interaction at some level. Even in AI-based techniques, humans interaction is involved either for ranking requirements, in the rule-making process, or the main decision-making process. We selected the techniques with the following properties:

- Techniques following Multiple criteria decision-making process
- Involve user interaction in decision making
- Techniques that deal with quantitative data
- Techniques that handle both functional and non-functional requirements
- Techniques that can handle a larger dataset

We represented non-functional requirements (dependencies of requirements) in terms of quantitative data. For example, a functional requirement calls “door lock ”, if it depends on “locking through fingerprints”, it should depend on “fingerprints capacity”, “Alarms after wrong inputs”, “set privileges for home admin”, etc. We measured these dependencies (non-functional requirements) for each requirement in quantitative form from 1-9. The selected techniques from the literature that fulfill the above-mentioned criteria are AHP, ANP, Fuzzy-AHP, Fuzzy-ANP and interactive genetic algorithm (IGA).

Dependent and Independent Variables

Our empirical study has three dependent variables (i.e., accuracy, efficiency and scalability of the techniques). Our independent variables are the requirements prioritization techniques (AHP, ANP, FAHP, FANP, and IGA).

- For measuring accuracy, we compared the results of the techniques to the baseline data, the more the results showed similarity to the baseline the more was the accuracy value (Mczara et al., 2014; Mustafa & Zainuddin, 2014).
- For measuring efficiency, we measured the effort in terms of time that a specific technique took to complete the prioritization task (Mczara et al., 2015; Mustafa & Zainuddin, 2014).
- For measuring scalability, we measured the accuracy and the efficiency of the techniques for different scales of requirements (i.e., 30, 60 and 100).

Data Analysis

We analyzed the data using statistical analysis. The tests applied were dependent on the nature of the data (i.e., the normality of the data, data dependency, and the size of the sample). We applied the statistical tests on the data that we gathered after: (i) practically implementing the selected techniques, (ii) feeding data in episodes to the codes written for each technique, (iii) and recording the values of time taken and similarity to the baseline index. Then we checked the normality of the data to check whether we met the assumption that the data is not normal. The tests that we applied for analyzing the hypothesis related to accuracy were the "Wilcox signed-rank test" and for identifying the effect size we used Cohen's D (CD).

Results and Discussion

In this section, we present and discuss the results of the analysis performed for the dependent variables (i.e., accuracy and efficiency for a different number of requirements). First, we checked the normality of the data by applying tests such as qqnorm, qqline, and Shapiro test. The resulting p-value was lesser than the significant value that is .05. The p-values for all the techniques are shown in Table 2 (columns 2, 4, and 6). As the data is not normal and for checking the homogeneity of variance, we applied the "Ansari" and "Mood" test which was true for AHP, ANP, and IGA while rejecting for FAHP and FANP. The p values of the homogeneity tests for the techniques are (.40, .38, .8, 1.16e-07 and 2e-05).

Accuracy

According to the assumption that we use the "same dataset" for all the techniques and the statistical tests for "normality of the data" shows that our data is "paired" and "non-parametric". Therefore, the Wilcoxon Signed Rank test was run to test the null hypothesis (i.e., there is no difference in terms of accuracy among AHP, ANP, FAHP, and FANP). By applying the test for all the selected techniques the results showed that there was a difference among the selected techniques in terms of accuracy, which means that the test rejects the null hypothesis as the p-value is less than .05

We identified the exact amount of difference by applying Cohen's-D (CD) which defines the effect size such as small (.2), medium (.5) and large (.8) while interpreting the effect of an intervention. The values of Cohen's-D test for all the selected techniques are shown below in Table 2 (columns 3, 5, and 7).

Table 2 shows that the p-value of FAHP was close towards the acceptance of the null hypothesis as it was almost equal to .05 which was the significant value for null hypothesis acceptance. Consequently, Cohen's D also showed the smaller effect size, and the difference from the average is almost negligible which makes this technique more accurate than the rest.

Table 2

Normalization And Cohen's D Values For Different Size Of Datasets

Techs	30 Requirements		60 Requirements		100 Requirements		Avg-Accuracy
	p	*CD*	*p*	*CD*	*p*	*CD*	
AHP	$p = 3.01e\text{-}11$	2.70	$p = 1.6e\text{-}11$	2.28	$p < 2.2e\text{-}16$	1.41	2.1
ANP	$p = 2.2e\text{-}16$	2.45	$p = 1.6e\text{-}11$	1.99	$p < 2.2e\text{-}16$	1.37	1.9
FAHP	$p = 0.049$	0.02	$p = 0.11$	0.1	$p = 0.003$	0.2	0.7
FANP	$p = 5.5e\text{-}07$	1.57	$p = 2.4e\text{-}8$	1.1	$p = 2.6e\text{-}13$	1.2	1.2
IGA	$p = 2.9e\text{-}11$	0.65	$p < 2.2e\text{-}16$	4.53	$p = 2.9e\text{-}11$	1.71	2.23

We also identified that the fuzzy RP techniques have a significant difference as compared to AHP, ANP, and IGA in terms of accuracy. The average difference of all the techniques in for accuracy is presented in the bar-chart shown in Figure 1. It shows the difference in the baseline in terms of accuracy. FAHP shows the lowest difference with the baseline. However, AHP and ANP show the highest difference in terms of accuracy as compared to the baseline.

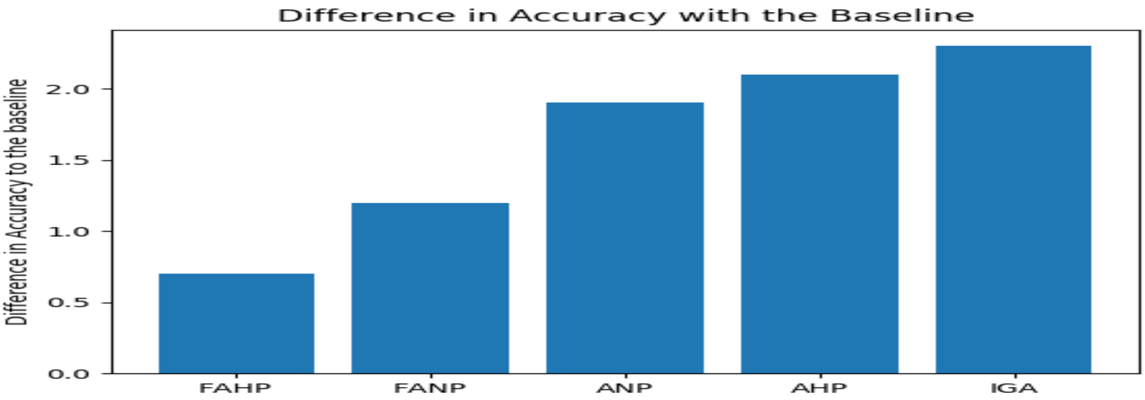

Figure 1. Bar-chart showing difference in the accuracy of the techniques

The accuracy value of all the techniques is shown in Table 2. The number of requirements is plotted on X-axis and the difference in accuracy is plotted on the Y-axis.

Efficiency

We measured the efficiency in terms of the total time that each technique took to complete a specific task. We present the time taken by each technique for all datasets of requirements in the form of a bar chart as shown in Figure 2, 3, and 4. For a dataset of 30 requirements, IGA was the most efficient technique while for a dataset of 60 requirements FAHP was more efficient as shown in Figure.3.

The efficiency results for a dataset of 60 requirements showed that FAHP was the most efficient technique and it remains effective for 100 requirements while IGA was unable to converge for 100 requirements as shown in Figure 4.

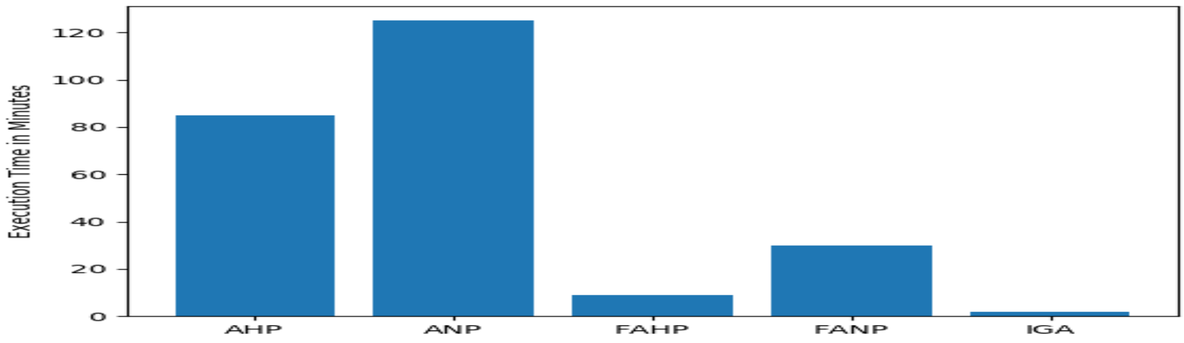

Figure 2. The efficiency of the techniques for 30 requirements

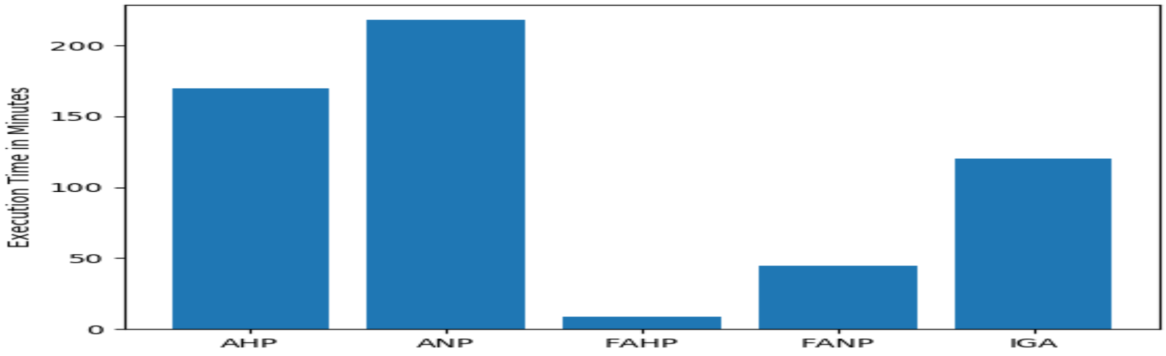

Figure 3. The efficiency of the techniques for 60 requirements

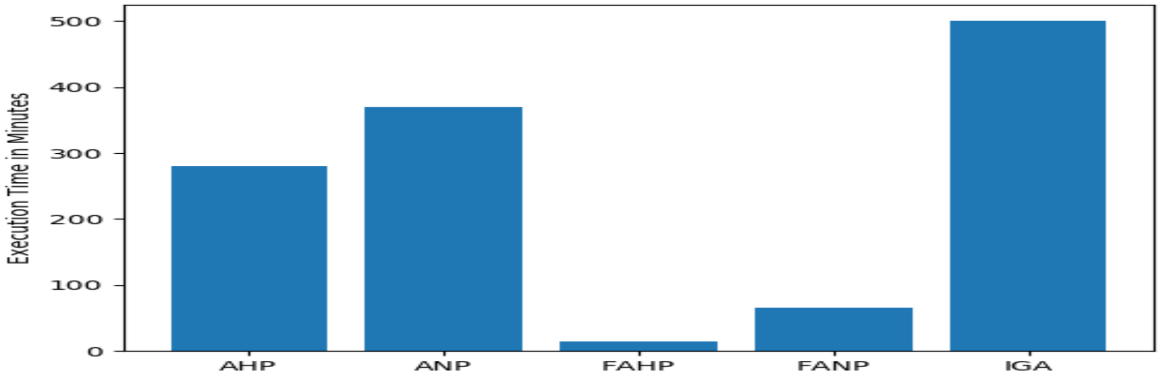

Figure 4. The efficiency of the techniques for 100 requirements

Results showed that the traditional techniques consume more time especially the ANP technique. ANP is a network process and contains more pair-wise comparisons that need more human effort to complete a prioritization task. We can see that FAHP is the most efficient technique in terms of time taken to prioritize the variable requirements sets as shown in the plot in Figure 3 and 4. The recorded time showed that IGA was the most time-consuming technique, especially for large datasets. There were more than 6,000,00 iterations of IGA for the large dataset (100 requirements). This makes IGA the most time-consuming technique especially for large data sets as compared with AHP, ANP, FAHP, and FANP.

Scalability

We measure scalability by measuring accuracy and efficiency for different scales of requirements. For measuring scalability, we tested the techniques for 30, 60 and 100 requirements. The reason for doing this was to explore the change in accuracy and efficiency values of the techniques for different size of requirements. We measured scalability in terms of high accuracy and less time (Hudaib, Masadeh, Qasem, & Alzaqebah, 2018). The more the technique is accurate and requires less time, the more scalable it is as compared to the rest. The results of accuracy and efficiency of the techniques are shown in Table 3. As presented in Table 3, FAHP has significantly higher accuracy different number of requirements while the other techniques showed lower accuracy with an increase in the size of the dataset. This showed that FAHP was highly scalable in terms of accuracy as well as it required less time to prioritize requirements as shown in Figure 3 and 4.

Table 3

Accuracy and Efficiency of Techniques for Variable Number of Requirements

Tech	30 requirements		60 requirements		100 requirements		Avg
	Accuracy	Efficiency (min)	Accuracy	Efficiency (min)	Accuracy	Efficiency (min)	Efficiency (min)
AHP	5.82	85	5.6	170	5.7	285	180
ANP	5.71	125	5.4	218	5.5	370	238
FAHP	0.02	9	0.1	9	0.2	9	9
FANP	1.57	30	1.1	30	1.2	30	30
IGA	0.65	2.19	4.53	120	3.7	***	***

The results of Cohen's D test showed that FANP is relatively less accurate than FAHP, IGA has very low accuracy while the AHP and ANP showed extremely low accuracy. The *** represents that IGA was unable to prioritize 100 requirements after being iterated for 5-7 times and running for 7 hours after which the system got stuck. Therefore, we were unable to identify the exact time taken by IGA to prioritize the dataset comprising of 100 requirements.

The line chart shown in Figure 5 presents the variation in efficiency of the techniques for different-sizes-of-data (scalability). The techniques that use fuzzy logic were the most efficient techniques compared to the rest of the techniques. Although the efficiency level varied for a different number of requirements still fuzzy-based approaches (i.e., FAHP and FANP outperformed the rest and handled variation in dataset size efficiently).

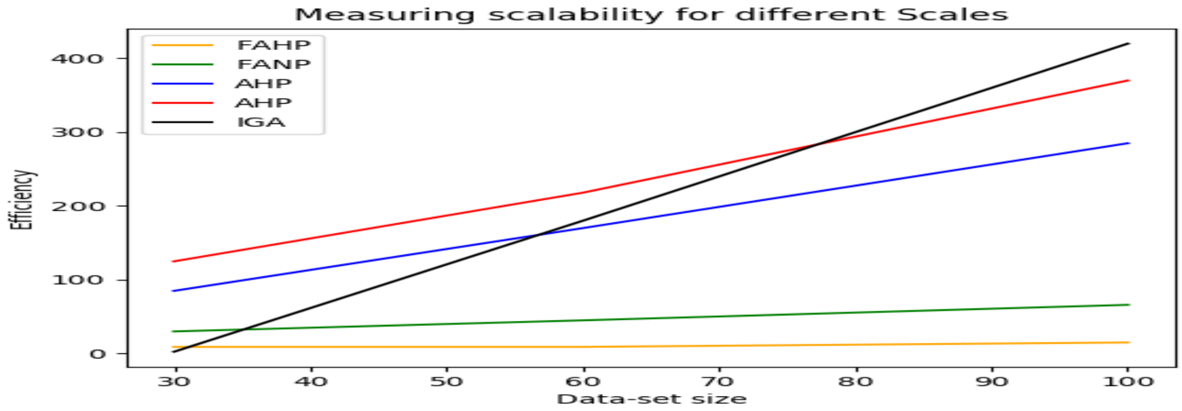

Figure 5. Scalability of the techniques in terms of efficiency

In summary, the results showed that AHP did not perform efficiently for different sizes of datasets as compared to other techniques as shown in Figure 2, 3, and 4. The findings also suggested that AHP was not scalable for a larger number of requirements (as shown in Figure 5. Our results corroborate the existing results of literature (Nidhra, Satish, & Ethiraj, 2012) in which AHP was compared with the Planning Game (PG) technique and the results showed that PG was efficient than AHP because AHP involved pair comparison which is time taking and requires more effort.

ANP showed a significant difference in terms of accuracy and efficiency for different number of requirements. However, ANP tuned out to be the most time-consuming technique (as shown in Figure 2, 3, and 4). ANP networks between the criteria and the alternative due to which it requires more pair-wise comparisons than AHP and consequently needs more effort and time than AHP's implementation. Due to its complexity, this technique was also not scalable for variable sizes of requirements.

FAHP proved to be the most accurate technique compared with the rest of the techniques. FAHP requires less effort to prioritize, hence it is efficient. Our findings corroborate the existing literature claims which say that FAHP enhances the efficiency of the prioritization technique (Gulzar et al., 2017). Moreover, FAHP was highly accurate and scalable when tested for a variable number of requirements as shown in the table. III.

FANP turned out to be the second most efficient and accurate technique as compared with the rest as shown in Figure 1 and Table 2. The lengthy process of creating clusters of the criteria for making a decision based on rules setting (Özdağoğlu, 2012) makes FANP the second most efficient and scalable technique as compared with the rest (as shown in Figure 6).

The results showed IGA as the most time-consuming technique, especially for a large number of requirements. Our results contradict with the previous study of (Tonella. 2010) in which IGA was showed as the most efficient technique for prioritizing forty requirements prioritization. However, we tested IGA for 100 requirements and it turned out to be a disaster because the algorithm iterated up to 600000 times and remained unable to terminate. The statistical results showed that IGA was the most time-consuming technique of the rest, even for 60 requirements. The random selection of the requirements for crossover and mutation in IGA makes it time-consuming (as shown in Table 3).

Threats to Validity

Internal Validity

Threats to internal validity are the uncontrolled factors that may possibly affect the results of treatment on the variables.

Participants: We ensured that the participants were already aware of requirements prioritization basics and we also guided them during the activity to rank requirements. However, the level of understanding participants had of the subject could vary. Since we did not conduct any pre-activity test to assess the subject grip of the students, thus variable levels of knowledge students had may affect their prioritization choices. Another important point is that this activity was conducted as a part of the course assignment which influenced the participants to complete the activity. This may have influenced some of the participants to rank requirements blindly. So there could be a rank bias towards specific criteria of requirements prioritization.

Task: Fatigue during the completion of the task may have discouraged the students to think properly and rank the requirements because there were 100 requirements and 4 prioritization factors that could have made the participants to make a lot of comparisons and think repeatedly.

External Validity

External validity is concerned with the generalization of the results outside the experimental environment.

Participants: The external validity threat is that we targeted the graduate students to rank the baseline dataset and did not consider the industrial participants. The results may have changed by using the industrial participants for requirements ranking.

Experimentation effect: By changing the participants from students to industrial practitioners, the accuracy, efficiency, and scalability of the techniques would be subject to change.

Conclusion

In this paper, we have presented an empirical study of requirement prioritization techniques. The aim was to explore the effect of the number of requirements on the accuracy and the efficiency of the prioritization techniques. We selected some AI-based techniques (i.e., AHP, ANP, FAHP, FANP, and IGA) and compared the results with a human prioritized list (called baseline) of requirements. The participants who participated in baseline creation were graduate students of software engineering course. Results showed that fuzzy logic-based techniques are more efficient and accurate than other techniques like AHP, ANP, and IGA. The statistical analysis showed that FAHP was the most scalable technique for a large dataset compared with other selected techniques.

Limitation and Future work

In this study, we prepared our baseline using academic setup and our participants were students. However, we aim to repeat the process with experts from the industry. The comparison could be used to identify the differences as well. In the future, we aim to extend our study by adding more features of requirements prioritization like dependency. Furthermore, we aim to extent our comparative analysis by empirically evaluating other AI-based techniques with fuzzy logic techniques for requirements prioritization for larger datasets.

References

Achimugu, P., Selamat, A., Ibrahim, R., & Naz, M. (2014). A systematic literature review of software requirements prioritization research. *Information and Software Technology, 56*(6), 568–585. https://doi.org/10.1016/j.infsof.2014.02.001

Agrawal, R., Singh, D., & Sharma, A. (2016, August). Prioritizing and optimizing risk factors in agile software development. Presented in *the Ninth International Conference on Contemporary Computing (IC3)* (pp. 1-7). IEEE.

Ahuja, H., & Batra, U. (2018). Performance enhancement in requirement prioritization by using least-squares-based random genetic algorithm. In *Innovations in Computational Intelligence* (pp. 251-263). Springer, Singapore.

Article, O. (2012). A multi-criteria decision-making methodology on the selection of facility location : fuzzy ANP, 787–803. https://doi.org/10.1007/s00170-011-3505-1

Dabbagh, M., Peck, S., Reza, L., & Parizi, M. (2016). Functional and non-functional requirements prioritization : empirical evaluation of IPA , AHP-based , and HAM-based approaches. *Soft Computing*, *20*(11), 4497–4520. https://doi.org/10.1007/s00500-015-1760-z

Daneva, M., Veen, E. Van Der, Amrit, C., Ghaisas, S., Sikkel, K., Kumar, R., … Wieringa, R. (2013). The Journal of Systems and Software Agile requirements prioritization in large-scale outsourced system projects : An empirical study. *The Journal of Systems and Software*, *86*(5), 1333–1353. https://doi.org/10.1016/j.jss.2012.12.046

Gulzar, K., Sang, J., Ramzan, M., & Kashif, M. (2017). Fuzzy approach to prioritize usability requirements conflicts : An experimental evaluation, *IEEE Access*, *5*, 13570–13577https://doi.org/10.1109/ACCESS.2017.2725321

Herrmann, A., & Paech, B. (2009). Practical challenges of requirements prioritization based on risk estimation, *Empirical Software Engineering*, *14*(6), 644–684. https://doi.org/10.1007/s10664-009-9105-0

Hudaib, A., Masadeh, R., Qasem, M. H., & Alzaqebah, A. (2018). Requirements prioritization techniques comparison. *Modern Applied Science, 12*(2), 62. https://doi.org/10.5539/mas.v12n2p62

Jawale, B. B., Patnaik, G. K., & Bhole, A. T. (2017). Requirement prioritization using adaptive fuzzy hierarchical cumulative voting. Presented in *the 7th International Advance Computing Conference (IACC)* (pp. 95-102). IEEE. https://doi.org/10.1109/IACC.2017.26

Mczara, J., Sarkani, S., Holzer, T., & Eveleigh, T. (2015). Software requirements prioritization and selection using linguistic tools and constraint solvers — a controlled experiment. *Empirical Software Engineering*, *20*(6), 1721–1761. https://doi.org/10.1007/s10664-014-9334-8

Mustafa, B. A., & Zainuddin, A. (2014, August). An experimental design to compare software requirements prioritization techniques. Presented in the *International Conference on Computational Science and Technology (ICCST)* (pp. 1-5). IEEE.

Nidhra, S., Satish, L. P. K., & Ethiraj, V. S. (2012). Analytical hierarchy process issues and mitigation strategy for large number of requirements. Presented in *CSI Sixth International Conference on Software Engineering (CONSEG)* (pp. 1-8). IEEE.

Özdağoğlu, A. (2012). A multi-criteria decision-making methodology on the selection of facility location: fuzzy ANP. *The International Journal of Advanced Manufacturing Technology*, *59*(5-8), 787–803.

Pergher, M., Massimilianopergherstud-infunibzit, E., & Rossi, B. (2013). Requirements prioritization in software engineering : A systematic mapping study. Presented in th *3rd International Workshop on Empirical Requirements Engineering (EmpiRE)* (pp. 40-44). IEEE.

Perini, A., Ricca, F., Bruno, F., Irst, K., & Susi, A. (2007). An empirical study to compare the accuracy of AHP and cbranking techniques for requirements prioritization. Presented in the *Fifth International Workshop on Comparative Evaluation in Requirements Engineering* (pp. 23-35). IEEE.

Sadiq, M., & Hassan, T. (2017). AHP _ GORE _ PSR : Applying analytic hierarchy process in goal oriented requirements elicitation method for the prioritization of software requirements. Presented in the *3rd International Conference on Computational Intelligence & Communication Technology (CICT)* (pp. 1-5). IEEE.

Tonella, P., Palma, F., Tonella, P., & Susi, A. (2010). Using interactive GA for requirements prioritization using interactive GA for requirements prioritization. Presented in the *2nd International Symposium on Search Based Software Engineering* (pp. 57-66). IEEE. https://doi.org/10.1109/SSBSE.2010.17

Tonella, P., Susi, A., & Palma, F. (2013). Interactive requirements prioritization using a genetic algorithm. *Information and Software Technology*, *55*(1), 173–187. https://doi.org/10.1016/j.infsof.2012.07.003

Vaidya, O. S., & Kumar, S. (2006). Analytic hierarchy process : An overview of applications, *European Journal of Operational Research*, *169*(1), 1–29. https://doi.org/10.1016/j.ejor.2004.04.028

Vestola, M. (2010). A comparison of nine basic techniques for requirements prioritization. *Helsinki University of Technology*, 1–8.