Using Fuzzy Linguistic 2-Tuples to Collectively Prioritize Software Requirements based on Stakeholders' Evaluations

Vassilis C. Gerogiannis
Dept. of Business Administration,
Technological Education Institute of Thessaly, 41110, Larissa, Greece
gerogian@teilar.gr

Georgios Tzikas
School of Science & Technology,
Hellenic Open University, 26335,
Patras, Greece
srd80802@ac.eap.gr

ABSTRACT

Efficient consideration of all stakeholders' needs and perspectives in a software project is a key challenge, especially when prioritizing the software requirements to be developed in the next software release. This paper presents a new requirements prioritization approach that aims to collectively prioritize software requirements based on their ratings expressed from different stakeholders. The proposed approach follows the steps of a value-oriented process in which multiple and possibly distributed stakeholders assess the values of candidate requirements with respect to various prioritization criteria. The approach applies a group-based, fuzzy multi-criteria technique requiring from involved stakeholders to evaluate requirements using linguistic terms. Stakeholders' linguistic evaluations are aggregated to collectively derive more objective and reasonable assessments on the final requirements' priorities.

CCS CONCEPTS

• **Software and its engineering** → Software creation and management → Designing software → Requirements analysis

KEYWORDS

Requirements prioritization, fuzzy linguistic 2-tuples, stakeholder management

1 INTRODUCTION

Requirements prioritization (RP) is an important decision making activity in requirements engineering that aims to rank candidate software requirements to be developed in the next software release according to stakeholders' needs and multiple cost, benefit and risk criteria [1]. Although there is a plethora of RP techniques, most of them are not widely adopted in practice mainly due to [2, 3]: (i) the scalability and computational complexity of their use, (ii) the insufficient consideration of project stakeholders, (iii) their sensitivity on subjective ratings of requirements as these are expressed by stakeholders' uncertain evaluations, and (iv) the inadequate consideration of dependencies which possibly exist among requirements. For example, RP techniques such as those based on pairwise comparisons (e.g., Analytic Hierarchy Process [4], cost - value approach [5] and bubble sort [6]) are often time and effort-consuming since require stakeholders to perform numerous pairwise comparisons between candidate requirements and their prioritization criteria. Techniques like numerical assignment [7], ranking of requirements [4] and cumulative voting on the requirements [8], according to a specific prioritization criterion (e.g., benefit, cost, risk, penalty), each time, often result in difficulties to consider all stakeholders' views and their evaluation uncertainty [2].

The problem of biased RP [9, 10] is also met in distributed software projects where involved stakeholders cannot easily negotiate and collaborate, being not at the same time/location, to reach a consensus when evaluating software requirements and, thus, RP results depend on discordant rates given by individuals who may even not have complete knowledge and understanding about project priorities. In addition, there are a few fuzzy-based RP techniques (e.g. [11, 12]) which try to handle the uncertainty of stakeholders' evaluations by requiring from them to express their ratings in linguistic terms and, consequently, by applying fuzzy set-based computations to derive requirements' priorities.

This paper presents a new RP approach that aims to collectively prioritize software requirements based on ratings expressed from different stakeholders. The proposed approach follows the steps of a value-oriented RP process [13] in which multiple stakeholders evaluate the value (impact) of candidate requirements on selected prioritization criteria. The approach applies a group-based, fuzzy multi-criteria technique which requires from involved stakeholders to evaluate candidate requirements using linguistic terms. The technique performs a similarity degree-based aggregation step upon all stakeholders' linguistic evaluations to collectively derive an aggregated and more objective assessment of the final requirements' priorities. This step computes the similarity degree between any two linguistic evaluations provided by any two stakeholders. Similarity degrees are used to adjust the weight of the rates of each stakeholder with the aim the final requirements ranking to reflect the collective/aggregated evaluation of all stakeholders more reasonably and more objectively.

The underlying representation/computation model of the technique is the fuzzy 2-tuple linguistic model (F2TL) [14]. The F2TL model follows the computing with words paradigm to combine numerical information and linguistic evaluations without any loss of information in the transformation process between numerical and linguistic values. The F2TL model has been widely employed in many evaluation and decision making problems where decision makers prefer to express their uncertain evaluations qualitatively [15] but, to the extent of our knowledge, the model has not been used to support the RP problem.

Section 2 presents the steps of the proposed RP approach along with a case study aimed to prioritize the requirements of a commercial company's marketing portal that provides registered company's customers with information about the company's products. This project executed in a distributed development context where the team of stakeholders involved in RP consisted of three individuals, namely the *product manager*, the *product marketing manager* and a *customer representative*, located, respectively, in three different geographical locations. Section 3 critically discusses the advantages and the limitations of the RP approach as well as briefly reviews the related literature. Section 4 concludes the paper and presents our future research directions.

2 DESCRIPTION OF THE RP APPROACH

In the following we describe the steps of the RP approach along with examples from a case study aimed to demonstrate the approach applicability.

Step 1: Specify Importance Values for Stakeholders and Determine Weights for the Prioritization Criteria. Assuming that a software project follows an iterative agile development approach [16], at the beginning of every development release, all stakeholders involved in RP (e.g., requirements analysts, designers, developers, customer representatives etc.) have to be rated according to criteria such as their experience, importance and influence on the project success. In addition, the RP criteria and their weights have to be determined.

The importance of each project stakeholder can be specified either implicitly, by asking each stakeholder to evaluate other stakeholders, or explicitly, by the product owner's decision. In case of a project with many stakeholders, implicitly defined importance values for stakeholders can be, for example, specified by asking each stakeholder to recommend other stakeholders [10]. Recommendations are represented in the form of a social network with stakeholders as nodes and recommendations as links. Social Network Analysis on the stakeholders' network can be used to calculate the importance of each stakeholder. In case of a project with a small number of stakeholders, importance values of stakeholders are often determined explicitly by the product owner. In agile projects, for example, the product owner may apply an approach such as AHP to compare stakeholders pairwise [17] or may construct a power-interest grid [18] to classify stakeholders, where high-importance stakeholders are those with both high power and high interest in the project.

Let us assume that, either implicitly or explicitly, a relative importance imp_sh_k for each one of K involved stakeholders in a software project sh_k $(k = \{1, \ldots, K\})$ has been determined $(\sum_{k=1}^{K} imp\_sh_k = 1)$. In the exemplar case study, in which we applied the RP approach to prioritize the requirements of a commercial company's marketing portal, three stakeholders are participated $(K = 3)$, namely, the *product owner* who has also the role of the *product manager* (sh_1), the *product marketing manager* (sh_2) and a *customer representative* (sh_3). We assume for each of them an equal relative weight of importance (i.e., $imp\_sh_1 = imp\_sh_2 = imp\_sh_3 = 1/3 = 0.333$).

Let us also assume that the product owner (often in cooperation with the most important stakeholders) defines M criteria for prioritizing candidate requirements cr_m ($m = \{1, \ldots, M\}$) and determines a corresponding weight w_cr_m for each prioritization criterion. Various criteria for RP have been proposed in the relevant literature [19], such as the importance/benefit of each requirement, the cost of requirement implementation, the penalty with not implementing a requirement, risk-related criteria, volatility-related criteria etc. Each one of these criteria is actually a category that includes a set of (sub-) criteria. E.g., the cost of implementation category includes criteria such as [19] (i) the complexity of a requirement, (ii) the level of experience related with the requirement implementation, (iii) the developer expertise diversity required for the requirement implementation, (iv) the effort required for the implementation, (v) the extend of documentation associated with a requirement and its implementation etc. When applying the proposed approach, it is recommended to select criteria of the same category.

All stakeholder evaluations can be expressed in terms of a linguistic term set $S = \{s_0, s_1, \ldots, s_g\}$ comprised by $g + 1$ terms. E.g., in the case study, the product owner selected the following set of seven (i.e., $g = 6$) linguistic terms, each one corresponding to a triangular fuzzy number: s_0 = *VVL* (*Very Very Low*) = (0,0,0.17), s_1 = *VL* (*Very Low*) = (0,0.17,0.34), s_2 = *L* (*Low*) = (0.17,0.34,0.5), s_3 = *M* (*Medium*) = (0.34,0.5,0.67), s_4 = *H* (*High*) = (0.5,0.67,0.84), s_5 = *VH* (*Very High*) = (0.67,0.84,1), and s_6 = *VVH* (*Very Very High*=(0.84,1,1). All linguistic evaluations are formulated as fuzzy 2-tuple linguistic terms (F2TL). Let $s_i \in S$ be a linguistic evaluation term. Based on the F2TL model, the following transformation function θ is used to result in the corresponding 2-tuple linguistic fuzzy information of s_i [14]:

$$\theta: S \rightarrow S \times [0.5, 0.5), \theta(s_i) = (s_i, 0), s_i \in S \quad (1)$$

The following four ($M = 4$) importance/benefit prioritization criteria are identified in the marketing portal case study, namely: cr_1: "urgency of implementing a requirement", cr_2: "contribution of a requirement to the portal acceptance", cr_3: "value of a requirement with regard to the portal users" and cr_4: "importance of a requirement with regard to the marketing position of the company". Their corresponding weights, as they are decided by the product owner, are expressed into 2-tuple linguistic terms as follows: $w\_cr_1 = (VH, 0)$, $w\_cr_2 = (VH, 0)$, $w\_cr_3 = (VVH, 0)$ and $w\_cr_4 = (VVH, 0)$.

It should be noted that stakeholders and the product owner may select different linguistic term sets (i.e., sets which do not have the same granularity or semantics) to express their evaluations. All heterogeneous linguistic evaluations should be initially transformed and expressed into a uniform linguistic term set following the method proposed in [20]. It should be also noted that the sensitivity of the final RP results is recommended to be examined using different input values for the importance values of the stakeholders and the weights of the prioritization criteria.

Step 2: Stakeholders Evaluate Requirements with respect to the Prioritization Criteria. Let us assume that in a specific development sprint of the software release N requirements are elicited to be prioritized. When applying RP, it is recommended [19] that all requirements needed to be prioritized should be selected from the same abstraction level e.g., goals, scenarios, solution oriented requirements, features, functional requirements, non-functional requirements etc. In the proposed approach, stakeholders use linguistic terms from a label set to evaluate the value of each elicited (at the same abstraction level) requirement R_n ($n = \{1, \dots, N\}$) with respect to each prioritization criterion cr_m ($m = \{1, \dots, M\}$). Each of these values, as it is given subjectively by each stakeholder sh_k ($k = \{1, \dots, K\}$), is denoted, in a 2-tuple form, as $(R_{n,m}^k, 0)$.

In a specific development sprint of the marketing portal case study, four functional requirements ($N = 4$) are elicited. Their descriptions, in a high-level form, are: R_1: "a registered customer uses the portal to issue an order for a company's product", R_2: "a non-registered customer uses to portal to download product brochures and catalogues", R_3: "a registered customer uses the portal to create and send to the company a direct mail asking information about a specific product" and R_4: "a registered customer uses the portal to write comments for a company's product". The values of these requirements with respect to the stated prioritization criteria, as they are provided by the four stakeholders involved in the case study, are shown in Table 1.

Step 3: Calculate Weighted Aggregated Ratings for the Values of Requirements. The input evaluation values given by the stakeholders for the requirements have to be aggregated by considering the importance value of each stakeholder. We have also to handle possible discordances in stakeholder evaluations. Thus, the final priority value of each requirement should reflect not only the importance of each stakeholder but also the similarities/agreements between stakeholders' evaluations. We assume that similar evaluations for the same requirement from different stakeholders should take higher weights when they are aggregated to compute the final requirements' priorities. In particular, the proposed RP approach applies similarity degree-based aggregation [21]. A similarity degree value $sim(R_{n,m}^k, R_{n,m}^l) \in [0,1)$ is calculated for any two evaluations $(R_{n,m}^k, 0)$ and $(R_{n,m}^l, 0)$ given respectively by stakeholders sh_k and sh_l ($k \neq l, k = \{1, \dots, K\}, l = \{1, \dots, K\}$) for the same requirement R_n ($n = \{1, \dots, N\}$) with respect to the same prioritization criterion cr_m ($m = \{1, \dots, M\}$). To calculate a similarity degree value, the distance δ between the corresponding $R_{n,m}^k$ and $R_{n,m}^l$ evaluations is computed. The evaluations, expressed in 2-tuple linguistic forms, should be transformed into equivalent numerical values. According to the F2TL model [14], given a linguistic term set $S = \{s_0, s_1, \dots, s_g\}$, $\beta \in [0, g]$ is a real number denoting the result of a symbolic aggregation operation. Let $i = round(\beta)$ (i.e., i is the integer value that results from applying the round operator on β) and $\alpha = \beta - i$ be two values where $i \in [0, g]$ and $\alpha \in [-0.5, 0.5)$, respectively. The number α is the result of the symbolic translation. The 2-tuple that denotes the equivalent information with the real number β results from the translation function $\Delta(\beta)$ as follows:

$$\Delta : [0, g] \rightarrow S \times [-0.5, 0.5)$$

$$\Delta(\beta) = (s_i, \alpha) = \begin{cases} s_i, i = round(b) \\ \alpha = \beta - i, \alpha \in [-0.5, 0.5) \end{cases} \quad (2)$$

A 2-tuple linguistic variable is transformed into an equivalent numerical value $\beta \in [0, g]$ by the reverse function Δ^{-1}:

$$\Delta : S \times [-0.5, 0.5) \rightarrow [0, g]$$

$$\Delta^{-1}(s_i, \alpha) = i + \alpha = \beta \quad (3)$$

Using the above (reverse) function (Δ^{-1}) the distance δ between evaluations $R_{n,m}^k$ and $R_{n,m}^l$ is calculated equal to:

$$\delta = \left| \Delta^{-1}(R_{n,m}^{\kappa}) - \Delta^{-1}(R_{n,m}^l) \right| \quad (4)$$

Consequently, the similarity degree value $sim(R_{n,m}^{\kappa}, R_{n,m}^l)$ is computed by the following equation, where g is the granularity of the used linguistic term set. The closer the similarity degree to 1, the more similar are the evaluations of any two stakeholders for the same requirement with respect to the same prioritization criterion.

$$sim(R_{n,m}^k, R_{n,m}^l) = 1 - \left| \frac{\Delta^{-1}(R_{n,m}^{\kappa}) - \Delta^{-1}(R_{n,m}^l)}{g} \right| \quad (5)$$

The similarity degrees calculated in the case study are shown in Table 2. For example, please notice in Table 1 that stakeholders sh_1 and sh_2 evaluate R_2 with respect to cr_1 as L and VL, respectively. By applying equation (4), the similarity degree between these two evaluations is equal to:

$$sim(R_{2,1}^1, R_{2,1}^2) = 1 - \left| \frac{\Delta^{-1}(R_{2,1}^1) - \Delta^{-1}(R_{2,1}^2)}{6} \right| = 1 - \left| \frac{2-1}{6} \right| = 0.8333$$

To compute the average similarity degree $SM_{n,m}^k$ and the relative similarity degree $RSM_{n,m}^k$ for each stakeholder sh_k ($k = \{1, \dots, K\}$) regarding the evaluation of requirement R_n ($n = \{1, \dots, N\}$) with respect to criterion cr_m ($m = \{1, \dots, M\}$), we use the following two equations:

$$SM_{n,m}^k = \frac{\sum_{l=1, l \neq k}^{K} sim(R_{n,m}^k, R_{n,m}^l)}{K - 1} \quad (6)$$

$$RSM_{n,m}^k = \frac{SM_{n,m}^k(sh_k)}{\sum_{l=1}^{K} SM_{n,m}^k(sh_l)} \quad (7)$$

E.g., for stakeholder sh_1, the average similarity degree $SM_{1,2}^1$ regarding the evaluation of R_1 with respect to cr_2 is equal to:

$$SM_{1,2}^1 = \frac{sim(R_{1,2}^1, R_{1,2}^2) + sim(R_{1,2}^1, R_{1,2}^3)}{3-1} = \frac{0.83333+1}{2} = 0.916667$$

In the same way, the values $SM_{1,2}^2$ and $SM_{1,2}^3$ are calculated equal to 0.833333 and 0.916667, respectively. Thus, for sh_1, the relative similarity degree $RSM_{1,2}^{1}$ regarding the evaluation of R_1 with respect to cr_2 is equal to:

$$RSM_{1,2}^{1} = \frac{SM_{1,2}^{1}}{SM_{1,2}^{1} + SM_{1,2}^{2} + SM_{1,2}^{3}} = \frac{0.916667}{0.916667 + 0.833333 + 0.916667} = 0.34375$$

All relative similarity degrees, as they are calculated in the case study, are shown in Table 3.

Next, we calculate the relative evaluation weight $w_{n,m}^{k}$ for each stakeholder sh_k ($k = \{1, \dots, K\}$) regarding the evaluation of the value of requirement R_n ($n = \{1, \dots, N\}$) with respect to criterion cr_m ($m = \{1, \dots, M\}$). The value $w_{n,m}^{k}$ is calculated by the following equation where both the relative weight of importance imp_sh_k of stakeholder sh_k and the corresponding relative similarity degree $RSM_{n,m}^{k}$ are considered:

$$w_{n,m}^{k} = \frac{imp\_sh_k \times RSM_{n,m}^{k}}{\sum_{l=1}^{K}(imp\_sh_l \times RSM_{n,m}^{l})} \quad (8)$$

In the case study, having assumed equal relative importance weights for the three stakeholders (i.e., $imp\_sh_1 = imp\_sh_2 = imp\_sh_3$=1/3), their relative importance weights have no effect on the above equation. E.g., the value $w_{1,2}^{1}$ is computed as follows:

$$w_{1,2}^{1} = \frac{imp_{sh_1} \times RSM_{1,2}^{1}}{\sum_{l=1}^{3}(imp_{sh_l} \times RSM_{1,2}^{l})} = \frac{(1/3) \times 0.34375}{(1/3) \times 0.34375 + (1/3) \times 0.3125 + (1/3) \times 0.343555} = 0.34375$$

The weights $w_{1,2}^{2}$ and $w_{1,2}^{3}$ for the evaluations of stakeholders sh_2 and sh_3 are calculated in the same way and they are equal to 0.3125 and 0.34375, respectively.

The weighted aggregation for all requirements' ratings is computed by utilizing the weighted average operator [14]. For a set of linguistic 2-tuples $\{(s_1, a_1), (s_2, a_2), \dots, (s_n, a_n)\}$ and their corresponding weights $\{w_1, w_2, \dots, w_n\}$, the 2-tuple weighted average operator $\bar{x}$ is defined as follows:

$$\bar{x} = \Delta\left(\frac{\sum_{i=1}^{n}(\Delta^{-1}(s_i, a_i) \times w_i)}{\sum_{i=1}^{n} w_i}\right) = \Delta\left(\frac{\sum_{i=1}^{n}(\beta_i \times w_i)}{\sum_{i=1}^{n} w_i}\right) \quad (9)$$

where β_i is calculated using the reverse function Δ^{-1}, as it is shown in equation (3). The final aggregated rating $FAR_{n,m}$ for requirement R_n ($n = 1, 2, \dots, N$) with respect to the prioritization criterion cr_m ($m = 1, 2, \dots, M$) is computed by applying the weighted average operator on the linguistic evaluations of the requirements using as weights the previously calculated relative evaluation weights for the respective evaluations:

$$FAR_{n,m} = \Delta\left(\frac{\sum_{l=1}^{K}(\Delta^{-1}(R_{n,m}^{l}) \times w_{n,m}^{l})}{\sum_{l=1}^{K} w_{n,m}^{l}}\right) \quad (10)$$

E.g., the final aggregated rating for R_1 with respect to cr_2 is computed as follows:

$$FAR_{1,2} = \Delta\left(\frac{\Delta^{-1}(R_{1,2}^{1}) \times w_{1,2}^{1} + \Delta^{-1}(R_{1,2}^{2}) \times w_{1,2}^{2} + \Delta^{-1}(R_{1,3}^{3}) \times w_{1,3}^{3}}{w_{1,2}^{1} + w_{1,2}^{2} + w_{1,2}^{3}}\right) = \Delta\left(\frac{1 \times 0.34375 + 0 \times 0.3125 + 1 \times 0.34375}{0.34375 + 0.3125 + 0.34375}\right) = \Delta(0.68675) = (1, -0.31) = (\mathrm{VL}, -0.31)$$

Final aggregated ratings for requirements are shown in Table 4.

Step 4: Calculate the Final Requirements' Priorities. The final priority $Prio\_R_n$ of each requirement R_n ($n = 1, 2, \dots, N$) is computed by applying the weighted average operator using as weights the corresponding numerical values of the linguistic evaluations of the weights of the prioritization criteria w_cr_m ($m = 1, 2, \dots, M$):

$$Prio\_R_n = \Delta\left(\frac{\sum_{m=1}^{M}(\Delta^{-1}(FAR_{n,m}) \times w\_cr_m)}{\sum_{m=1}^{M} w\_cr_m}\right) \quad (10)$$

In the case study, we have specified that the weights of the prioritization criteria are equal to: $w\_cr_1 = (VH, 0)$, $w\_cr_2 = (VH, 0)$, $w\_cr_3 = (VVH, 0)$, $w\_cr_4 = (VVH, 0)$. Thus, the final priority of R_1 ("a registered customer uses the portal to issue an order for a company's product") is equal to:

$$Prio\_R_1 = \Delta\left(\frac{5.4285 \times 5 + 0.6875 \times 5 + 5.6875 \times 6 + 3.75 \times 6}{5 + 5 + 6 + 6}\right) = \Delta(3.9638) = (4, -0.04) = (\mathrm{H}, -0.04)$$

The final requirements' priorities are shown in the last row of Table 4. These are: $Prio\_R_1 = (H, -0.04)$, $Prio\_R_2 = (M, +0.26)$, $Prio\_R_3 = (M, -0.33)$, $Prio\_R_4 = (L, +0.008)$.

3 DISCUSSION AND RELATED WORK

An interesting assertion from applying the RP approach in the case study is that ties do not appear in the final requirements ranking. Ties in requirements ranking is a problem often met in numerical assignment, requirements grouping/classification and in other RP approaches [22]. The final requirements' priorities differ each other, even slightly, and the final ranking highlights priority differences. One limitation of the approach is that it may require considerable time and effort from stakeholders to evaluate a large number of requirements. This limitation is met in many other RP approaches [2]. We, therefore, propose to apply the approach in agile iterative projects implemented in weekly sprints, where the number of selected requirements to be prioritized is often rather small [16]. To derive a more complete assessment for requirements' priorities, stakeholders can use the approach to prioritize candidate requirements according to different prioritization criteria (e.g., importance/benefit, cost of implementation, risk, volatility). Stakeholders can also use the approach to evaluate and prioritize other types of requirement artifacts, such as business goals, solution oriented requirements and non-functional requirements as well as possible interdependencies between functional and non-functional requirements [23]. To facilitate the application of the approach in a (distributed) software development project with many stakeholders we have developed a prototype of a web-based RP tool that is called Re_Prio (http://sprint.teilar.gr/re_prio). Re_Prio manages linguistic evaluations given by RP stakeholders, stores evaluations in a database and performs all required calculations. Re_Prio can be also used to check the sensitivity of RP results using different inputs for the importance values of stakeholders and the weights of prioritization criteria.

Existing RP techniques in the literature are classified according to the nominal, ordinal, interval and ratio scale that they use to present information of the requirements ranking [2]. The presented approach belongs to the ordinal scale taxonomy, as stakeholders express their judgements using the ordinal scale of a linguistic term set. As Buckley states [24] there are several reasons for using the ordinal scale instead of other scales, e.g., in problems with a large number of alternatives or criteria it might be easier to provide evaluations in an ordinal scale. The results of the approach are expressed into fuzzy linguistic 2-tuples and, thus, indicating the extent at which one requirement is considered more important than others. The approach presents some similarities with Value oriented Prioritization [13] and the Wiegers' method [25]. These techniques require from representative stakeholders to provide numerical values to evaluate the relative benefit of each requirement, the relative penalty if a requirement is not implemented, the relative cost of implementing each requirement and the relative risk associated with each requirement. In the proposed approach, these evaluations are given from multiple stakeholders in linguistic terms and they are collectively aggregated considering similarities in stakeholders' evaluations. Thus, the approach can be potentially used in group-based RP settings. Finally, there are negotiation RP approaches, such as Win-Win [26], which require stakeholders to rank requirements privately before starting negotiations for the final priorities. Our approach considers discordances in stakeholders' evaluations and the final aggregated ranking can be collaboratively examined by stakeholders to reach an agreement on the final priorities.

4 CONCLUSIONS AND FUTURE WORK

We have presented a RP approach that aims to collectively prioritize software requirements based on uncertain ratings expressed from multiple stakeholders in linguistic terms. Similarities in stakeholders' ratings are used to derive a final requirements ranking that reflects the collective/aggregated evaluations of all stakeholders on the candidate requirements more reasonably, fairly and objectively. The proposed approach is based on qualitative stakeholders' evaluations expressed and calculated in the form of fuzzy linguistic 2-tuples. Our future research directions are twofold: (i) to validate the approach in a distributed software project with many stakeholders by utilizing the Re_Prio tool, and (ii) to further enhance the approach allowing stakeholders to use more than one linguistic terms to express their evaluations. It would be interesting to allow a stakeholder to use a linguistic evaluation in the form "between medium and very high" (instead of "medium", "high" or "very high") to express, in a more uncertain way, the value of a requirement with respect to a prioritization criterion. To cope with this extension we plan to extend the approach using Hesitant Fuzzy Linguistic Term Sets [27].

REFERENCES

[1] N.R. Mead. 2006. *Requirements Prioritization Introduction*. Software Engineering Institute Web Publication, Carnegie Mellon University, Pittsburgh.
[2] P. Achimugu, A. Selamat, R. Ibrahim, and M. N. Mahrin. 2014. A Systematic Literature Review of Software Requirements Prioritization Research. *Information and Software Technology*, 56(6), 568–585.
[3] R. B. Svensson, T. Gorschek, B. Regnell, R. Torkar, A. Shahrokni, R. Feldt, and A. Aurum. 2011. Prioritization of Quality Requirements: State of Practice in Eleven Companies. In *Proceedings of the 19th IEEE International Requirements Engineering Conference* (RE), IEEE, 69-78.
[4] J. Karlsson, C. Wohlin, and B. Regnell. 1998. An Evaluation of Methods for Prioritizing Software Requirements. *Information and Software Technology*, 39(14), 939–947.
[5] J. Karlsson, and K. Ryan. 1997. A Cost-Value Approach for Prioritizing Requirements. *IEEE Software*, 14(5), 67-74.
[6] M. Babar, M. Ramzan, and S. Ghayyur. 2011. Challenges and Future Trends in Software Requirements Prioritization. In *Proceedings of the Computer Networks and Information Technology* (ICCNIT 2011), IEEE, 319–324.
[7] P. Berander, and A. Andrews. 2005. Requirements Prioritization. In *Engineering and Managing Software Requirements*, A. Aurum and C. Wohlin (Eds.). Springer, 69-94.
[8] D. Leffingwell, and D. Widrig. 2003. *Managing Software Requirements: A Use Case Approach*. 2nd ed. Addison-Wesley, Boston.
[9] L. Lehtola, M. Kauppinen, and S. Kujala. 2004. Requirements Prioritization Challenges in Practice. In *Product Focused Software Process Improvement*. Springer, 497-508.
[10] S. L. Lim, and A. Finkelstein. 2012. StakeRare: Using Social Networks and Collaborative Filtering for Large-Scale Requirements Elicitation. *IEEE Transactions on Software Engineering*, 38(3), 707-735.
[11] A. Ejnioui, C. Otero, and A. Qureshi, 2012. Software Requirement Prioritization Using Fuzzy Multi-Attribute Decision making. In *Proceedings of the Open Systems Conference* (ICOS), IEEE, 1-6.
[12] D. Lima, F. Freitas, G. Campos, and J. Souza. 2011. A Fuzzy Approach to Requirements Prioritization. In *Search Based Software Engineering*, Springer, 64-69.
[13] J. Azar, R. K. Smith, and D. Cordes. 2007. Value-Oriented Requirements Prioritization in a Small Development Organization. *IEEE Software*, 24(1), 32-27.
[14] F. Herrera, and L. Martinez. 2000. A 2-Tuple Fuzzy Linguistic Representation Model for Computing with Words. *IEEE Transactions on Fuzzy Systems*, 8(6), 746-752.
[15] L. Martinez, and F. Herrera. 2012. An Overview on the 2-Tuple Linguistic Model for Computing with Words in Decision Making: Extensions, Applications and Challenges. *Information Sciences*, 207(10), 1-18.
[16] K. Schwaber. 2004. *Agile Project Management with Scrum*. Microsoft Professional.
[17] T. L. Saaty. 1980. *The Analytical Hierarchy Process*, McGraw-Hill.
[18] F. Ackermann, and C. Eden. 2011. Strategic Management of Stakeholders: Theory and Practice. *Long Range Planning*, 44, 179-196.
[19] K. Pohl. 2010. Requirements Engineering, Fundamentals, Principles, and Techniques. Springer.
[20] F. Herrera, L. Martinez, and P. J. Sanchez. 2005. Managing Non-Homogeneous Information in Group Decision Making. *European Journal of Operational Research*, 166 (1), 115-132.
[21] X. Liao, Y. Li, and B. Lu. 2007. A Model for Selecting an ERP System based on Linguistic Information Processing. *Information Systems*, 32(7), 1005-1017.
[22] S. Padmanabhuni, B. Boehm, S. S. Payyavula, and N. Kukreja. 2012. Selecting an Appropriate Framework for Value-based Requirements Prioritization. In *Proceedings of the 20th IEEE International Requirements Engineering Conference* (RE), IEEE, 303-308.
[23] M. Dabbagh, and S. P. Lee. 2014. An Approach for Integrating the Prioritization of Functional and Nonfunctional Requirements. *The*

Scientific World Journal, article ID 737626, 13 pages.
[24] J. J., Buckley. 1984. The Multiple Judge, Multiple Criteria Ranking Problem: A Fuzzy Set Approach. *Fuzzy Sets and Systems*, 13(1), 25–37.
[25] K.E., Wiegers. 1999. First Things First: Prioritizing Requirements. *Software Development*, 7(9), 48-53.
[26] P. Gruenbacher. 2000. Collaborative Requirements Negotiation with Easy Win Win. In *Proceedings of the 2nd International Workshop on the Requirements Engineering Process*, IEEE.
[27] R. M. Rodriguez, L. Martinez, and F. Herrera. 2012. Hesitant Fuzzy Linguistic Term Sets. *IEEE Transactions on Fuzzy Sets and Systems*, 20(1), 109-119.

Table 1: Stakeholders' Evaluations of the Requirements

Stakeholder / Relative Weight of Importance	Criterion	Evaluation of Requirements with respect to Criteria			
		R_1	R_2	R_3	R_4
$sh_1/(1/3)$	cr_1	$(H,0)$	$(L,0)$	$(VL,0)$	$(VVH,0)$
	cr_2	$(VL,0)$	$(VL,0)$	$(VH,0)$	$(VL,0)$
	cr_3	$(VH,0)$	$(VH,0)$	$(L,0)$	$(L,0)$
	cr_4	$(L,0)$	$(VVH,0)$	$(VL,0)$	$(VL,0)$
$sh_2/(1/3)$	cr_1	$(VVH,0)$	$(VL,0)$	$(M,0)$	$(VH,0)$
	cr_2	$(VVL,0)$	$(VL,0)$	$(VVH,0)$	$(L,0)$
	cr_3	$(VVH,0)$	$(M,0)$	$(L,0)$	$(VL,0)$
	cr_4	$(H,0)$	$(VH,0)$	$(VL,0)$	$(VL,0)$
$sh_3/(1/3)$	cr_1	$(VVH,0)$	$(VVL,0)$	$(M,0)$	$(M,0)$
	cr_2	$(VL,0)$	$(L,0)$	$(VH,0)$	$(L,0)$
	cr_3	$(VVH,0)$	$(VVH,0)$	$(L,0)$	$(VL,0)$
	cr_4	$(VH,0)$	$(VH,0)$	$(L,0)$	$(VVL,0)$

Table 2: Similarity Degree Values between Stakeholders' Evaluations

Criterion	Similarity between evaluations of sh_1 and sh_2				Similarity between evaluations of sh_1 and sh_3				Similarity between evaluations of sh_2 and sh_3			
	R_1	R_2	R_3	R_4	R_1	R_2	R_3	R_4	R_1	R_2	R_3	R_4
cr_1	0.666667	0.833333	0.666667	0.833333	0.666667	0.666667	0.666667	0.5	1	0.833333	1	0.666667
cr_2	0.833333	1	0.833333	0.833333	1	0.833333	1	0.8333333	0.833333	0.833333	0.833333	1
cr_3	0.833333	0.666667	1	0.833333	0.833333	0.833333	1	0.8333333	1	0.5	1	1
cr_4	0.666667	0.833333	1	1	0.5	0.833333	0.833333	0.833333	0.833333	1	0.833333	0.833333

Table 3: Relative Similarity Degree Values for Stakeholders' Evaluations

Criterion	sh_1				sh_2				sh_3			
	R_1	R_2	R_3	R_4	R_1	R_2	R_3	R_4	R_1	R_2	R_3	R_4
cr_1	0.285714	0.321429	0.285714	0.33333	0.357143	0.357143	0.357143	0.375	0.357143	0.321429	0.357143	0.291667
cr_2	0.34375	0.34375	0.34375	0.3125	0.3125	0.34375	0.3125	0.34375	0.34375	0.3125	0.34375	0.34375
cr_3	0.3125	0.375	0.33333	0.3125	0.34375	0.291667	0.33333	0.34375	0.34375	0.33333	0.333333	0.34375
cr_4	0.291667	0.3125	0.34375	0.34375	0.375	0.34375	0.34375	0.34375	0.33333	0.34375	0.3125	0.3125

Table 4: Aggregated Ratings and Final Priorities of the Requirements

Criterion / Importance	Aggregated Evaluation of the Requirements with respect to the Prioritization Criteria			
	R_1	R_2	R_3	R_4
cr_1 / $(VH,0)$	$5.4285/(VH,0.43)$	$1/(VL,0)$	$2.4285/(L,0.43)$	$4.75/(VH,-0.25)$
cr_2 / $(VH,0)$	$0.6875/(VL,-0.31)$	$1.3125/(VL,0.31)$	$5.3125/(VH,0.31)$	$1.6875/(L,-0.31)$
cr_3 / $(VVH,0)$	5.6875 / $(VVH,-0.31)$	$4.75/(VH,-0.25)$	$2/(L,0)$	$1.3125/(VL,0.31)$
cr_4 / $(VVH,0)$	$3.75/(H,-0.25)$	$5.3125/(VH,0.31)$	$1.3125/(VL,0.31)$	$0.6875/(VL,-0.31)$
Final Requirements' Priorities	$3.9638/(H,-0.04)$	$3.2698/(M,+0.26)$	$2.6627/(M,-0.33)$	$2.0085/(L,+0.008)$