A linguistic software requirement prioritization model with heterogeneous information.

Karina Pérez-Teruel[1] Maikel Leyva-Vázquez[2] Macarena Espinilla[3]

[1]University of Information Science, Highway to San Antonio, Havana, Cuba, karinapt@uci.cu
[2] University of Information Science, Highway to San Antonio, Havana, Cuba, karinapt@uci.cu
[3]University of Jaén. Department of Computer Science, Campus Las Lagunillas s/n. Jaén. CP. 23071 (Spain)
mestevez@ujaen.es

Abstract

Software engineers are involved in complex decisions that require multiples viewpoints. A specific case is the requirement prioritization process. Criteria involved in this process can be of different nature and results should be easily understandable. In this paper a software requirement prioritization model is develop based on the linguistic decision analysis scheme. The proposal can manage different types of information and the final results are expressed into a linguistic domain with the aim of facilitating understanding, following the computing with words paradigm. Finally, an illustrative example is provided to show the applicability of the proposed model.

Keywords: requirement engineering, software requirement prioritization, computing with words, decision analysis

1. Introduction

Software quality is influenced by the ability to satisfy client and user needs obtained and described in software requirements [1]. Many models have been proposed for software requirement prioritization [1-7]. However, these proposals present a lack of dealing with criteria of different nature.

Moreover results expressed quantitatively may be difficult to understand for software engineers. In order to overcome the drawbacks identified in this contribution we propose a novel and flexible requirement prioritization process in which experts can provide their judgments with different domains (numerical and linguistic).

For one hand, the heterogeneous information is unified into a linguistic domain, using the linguistic 2-tuple representation model [8], developing processes of computing with words (CW) [9] and providing linguistic results. Furthermore, the proposed model includes a two-step aggregation that uses the weighted average (WA) and the ordered weighted averaging (OWA) for linguistic 2-tuple. The ordered weighted averaging – weighted average (OWAWA) [10] in 2-tuple (2-TOWAWA) provides a more flexible representation of the WA and the OWA operators since it considers the degree of importance for each concept and includes them as particular cases. For the other hand, results of the software requirement prioritization process are provided by linguistic assessments because the linguistic domain is closed to human cognitive model and easily interpreted by software developers.

In software requirement prioritization are implied different stakeholders approaching to the decision problem from different angles. It is moreover a multidimensional problems dealing with multiple criteria of diverse nature. Therefore, the proposed model is based on a decision analysis scheme [11] and the approach presented in [8] in order to deal with heterogeneous information provided by several experts.

This paper is structured as follows: Section 2 outlines a scheme of linguistic decision analysis. Section 3 shows an extension of the linguistic 2-tuple to heterogeneous contexts. Section 4 presents our linguistic model for software requirements prioritization. Section 5 shows an illustrative example of the proposed model. The paper ends with conclusions and further work recommendations in Section 6.

2. Software requirement prioritization and linguistic decision analysis.

One frequent reason that cause low quality software is associated to problems related to identifying and selecting the most important requirements [12]. Software requirement prioritization can be modeled as a decision making problem helping decision maker to reach a consistent decision [13].

Due to this fact, our proposal for a linguistic software requirement prioritization model is based on the classical decision analysis scheme [11]. In this contribution, the software requirement prioritization process is modeled as a Multi-Expert Multi-Criteria decision making problem due to the complexity of the problem where multiple criteria and multiple experts are involved [12, 14].

In the prioritization process are involved quantitative criteria easily evaluated in a numerical way. However, there are some criteria that present subjectivity, vagueness and inaccuracy. These criteria should be evaluated in a qualitative way. Additionally for experts in the software requirement prioritization process sometimes could be difficult to express reality in a quantitative way. Fuzzy set theory, introduced by Zadeh [15] in 1965, offers a mathematical model to deal with this kind of uncertainty. The fuzzy linguistic approach is based in the fuzzy set theory and especially in the linguistic variable concept [16, 17]. The use of linguistic variables implies the need of Computing with Words (CW) [18] that considers that inputs and output results should be expressed in a linguistic domain to be close to natural language and provide interpretable and understandable results to humans. This fact is important in software requirement prioritization where evaluation results are used to make decisions by software engineers in high complexity environment [19].

3. Extension of the linguistic 2-tuple to manage heterogeneous contexts.

Criteria for software requirements prioritization may have different nature (quantitative and qualitative). Therefore, it is appropriate to express each criterion in the adequate domain (numerical or linguistic), generating a heterogeneous context. In this context, the extension of the linguistic 2-tuple model proposed in [8] is a good option because it provides linguistic results and has a low cost to include or remove an evaluated software requirement [20].

The linguistic representation model based in 2-tuples was proposed in [21] and defines a set of transformation functions for linguistic 2-tuple in order to carry out the CW process without loss of information. Being $\beta \in [0, g]$ a value that represents the result of a symbolic operation, a linguistic 2-tuple $(s_i, \propto)$ can be assigned in order to express the equivalent information of that given by β, where $s_i \in = \{s_0, \dots, s_g\}$ is a linguistic term and $\propto$ is a numerical value representation of the symbolic translation.

Definition 1.[21] Let $S = \{s_0, \dots, s_g\}$ be a set of linguistic terms. The 2-tuple set associated with S is defined as $\langle S \rangle = S \times [-0.5, 0.5)$. We define the function $\Delta: [0, g] \rightarrow \langle S \rangle$ given by,

$$\Delta(\beta) = (s_i, \propto), \ with \begin{cases} s_i, i = round(\beta) \\ \propto = \beta - i, \end{cases} \quad (1)$$

where round assign to β the integer number $i \in \{0, 1, \dots, g\}$ closest to β.

We note that Δ function is bijective [21] and $\Delta^{-1}: [0, g] \rightarrow S \times [-0.5, 0.5)$ is defined by $\Delta^{-1}(s_i, \propto) = i + \propto$.

Numerical values can be transformed to the linguistic domain (S_T) following a two step process. First transforming numerical values in [0, 1] to $f(S_T)$ using the following numerical linguistic transformation function.

Definition 2.[8] Let $v \in [0,1]$ be a numerical value and $S_t = \{s_0, s_1, \dots, s_g\}$ a linguistic terms set. The numerical linguistic transformation function $\tau NS_t: [0,1] \rightarrow F(S_t)$ is defined by:

$$\tau NS_t(v) = \{(s_0, \gamma_0), (s_1, \gamma_1), \dots, (s_g, \gamma_g)\} \text{ with}$$

$$\gamma_i = \mu_{s_i}(v) = \begin{cases} 0, if v < a \ or \ v > d, \\ \frac{v-a}{b-a}, if a < v < b, \\ 1, if b \leq v \leq c, \\ \frac{d-v}{d-c}, if c < v < d \end{cases} \quad (2)$$

Where $\gamma_i \in [0,1]$ and $F(S_t)$ is the set of fuzzy sets in S_t and $\mu_{s_i}(\cdot)$ is the membership function of the linguistic label $s_i \in S_t$. That is defined by a parametric function (a, b, d, c) [8].

The previous information unified into fuzzy sets in S_t is later transformed to facilitate the interpretability of the results. This transformation is conducted by the function $\chi: F(S) \rightarrow [0, g]$:

Definition 3.[8] Given the linguistic terms set $S_t = \{s_0, s_1, \dots, s_g\}$ the function $\chi: F(S) \rightarrow [0, g]$ is defined by,

$$\chi: (F(S_t)) = \chi(\{(s_j, \gamma_j), j = 0, \dots, g\}) = \frac{\sum_{j=0}^{g} j\gamma_j}{\sum_{j=0}^{g} \gamma_j} = \beta \quad (3)$$

where the fuzzy set $F(S_t)$ could be obtained from τNS_t (2). Applying the function Δ to β (1), we can assign a 2-tuple that expresses the equivalent information of that given by β.

There is a set of aggregation functions that operate with the linguistic 2-tuples, among them the 2-tuple weighted average (2-TWA) operator and the 2-tuple ordered weighted averaging (2-TOWA) operator [21]. The 2-tuple ordered weighted averaging weighted average (2-TOWAWA) operator is based on the ordered weighted averaging weighted averaging (OWAWA) operator [22]. It unifies the 2-TWA and the 2-TOWA operators in a single formulation. It can be defined as follows:

Definition 4. Let $A = \{(r_1, \alpha_1), \dots, (r_n, \alpha_n)\}$ be a vector of linguistic 2-tuple, and W a weighting vector of dimension n such as $\sum_{j=1}^{n} w_j = 1$ and $w_j \in [0,1]$, and a weighting vector V of dimension n that affects the WA operator, with $\sum_{i=1}^{n} v_i = 1$ and $v_i \in [0,1]$, such that:

$$2 - \mathrm{TOWAWA}((r_1, \alpha_1), \dots, (r_n, \alpha_n)) = \Delta(H \sum_{j=1}^{n} w_j \beta_j^* + (1 - H) \sum_{j=1}^{n} w_j \beta_j) \quad (4)$$

Where β_j^* is the jth largest element of the $\{\Delta^{-1}(r_1, \alpha_1), \dots, \Delta^{-1}(r_n, \alpha_n)\}$ and $H \in [0,1]$.

The 2-TOWAWA operator unifies each concept (2-TWA and 2-TOWA), in a more flexible way considering the degree of importance that that each of them has in the aggregation [23].

4. A linguistic software requirement prioritization model with heterogeneous information.

Our aim is to develop a software requirement prioritization model based on the linguistic decision analysis scheme that can deal with criteria with different nature and can provide linguistic results. The model consists of the following phases (graphically, Figure 2):

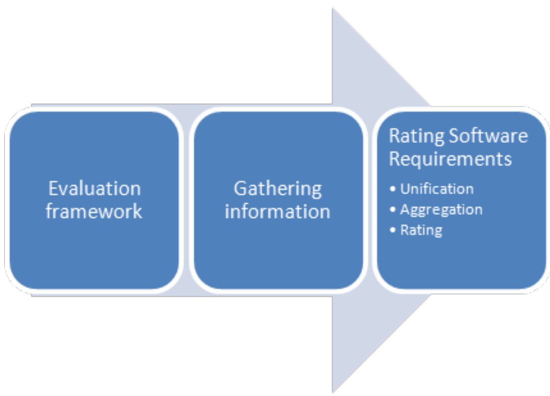

Figure 1. Scheme of the Model.

A. Evaluation framework:

In this phase, the evaluation framework is defined to fix the requirement prioritization problem structure. The framework is established as follows:

- Let E= $\{e_1, e_2, \dots, e_n\}$ $(n \geq 2)$ be a set of experts.
- Let C=$\{c_1, c_2, \dots, c_k\}$ $(k \geq 2)$ be a set of criteria.
- Let R=$\{r_1, r_2, \dots, r_m\}$ $(m \geq 2)$ be a set of requirements.

Here, we consider a heterogeneous information framework [8]. Each expert can use different domain (numerical or linguistic) to asses each criteria, attending to its nature.

B. Gathering information:

Once the framework has been defined, the knowledge of the set of experts must be obtained. Each expert provides their preferences by using utility vectors. The utility vector [24] is represented in the following way:

- $P_j^i = \{p_{j1}^i, p_{j2}^i, \dots, p_{jh}^i\}$,

where p_{jk}^i is the preference provided to the criterion c_k of the requirement r_j by the expert e_i.

C. Rating software requirements.

The aim of this phase is to obtain a collective linguistic global assessment easily interpretable for software engineers. To do so the information is unified and aggregated. Finally those requirements more prioritized are identified. This phase in based the approach reviewed in the Section 3 to deal with heterogeneous information and to give linguistic results.

Unification of the information

The numerical information is unified in the linguistic domain (S_T).

a) Transforming numerical values in [0, 1] to $f(S_T)$ (2).

b) Transforming fuzzy sets over the S_T into linguistic 2-tuple by equations (3) and (1).

Aggregation of the information

A two-step aggregation process is developed with the aim of compute a global evaluation. We obtain for each expert an assessment for each requirement. For computing these assessments we propose a 2-tuple OWAWA (2-TOWAWA) (4) operator, computing a collective evaluation taking into account optimism/pessimism degree and the importance of each criterion.

The final aim is to obtain a global evaluation of each requirement according to all experts. To do so, this process will aggregate all the experts' collective assessment by using the 2-tuple weighted average operator (2-TWA) [21]. We propose this operator to establish different weights for each expert, taking into account their knowledge and their significance in software requirement prioritization process

Rating of the requirements

The final step is to establish a ranking among software requirements; this ranking allows selecting the requirements with more value and postponing or rejecting the development of others making more effective the software development process. The best requirement is the one with the maximum collective evaluation $Max\{(r_j, \alpha_j), = 1, 2, \dots, m\}$. Requirements are prioritized based on this value in a decreasing order.

5. Illustrative Example

In this section, we present an illustrative example in order to show the applicability of the proposed model.

A. Evaluation framework

In this case study the evaluation framework is compose by: 3 experts $E = \{e_1, e_2, e_3\}$, who evaluate 3 requirements $R = \{r_1, r_2, r_3\}$, where are involved 5 criteria $C = \{c_1, c_2, \dots, c_5\}$, which are shown below:

- c_1: Importance for the customers
- c_2: Value
- c_3: Cost
- c_4: Technical Complexity
- c_5: Risks

Each expert could give the information in a numerical or linguistic way attending to the nature of the criteria. The linguistic results also will be expressed in this domain, S_t (Figure 2).

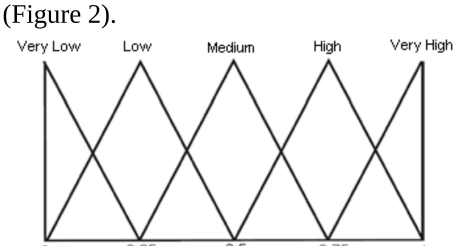

Figure 2. Selected domain S_t.

B. Gathering information

Once the evaluation framework has been determined the information about the requirements is gathered (see Table I). Qualitative criteria will be evaluated in the S_t scale.

Table I. An illustrative example of gathering information.

	e_1			e_1			e_1		
	r_1	r_2	r_3	r_1	r_2	r_3	r_1	r_2	r_3
c_1	VH	H	VH	VH	H	H	M	VH	H
c_2	H	H	L	M	VH	H	H	H	H
c_3	0.9	0.7	0.6	0.9	0.9	0.6	0.6	0.9	0.4
c_4	H-	M	H	VH	L	VH	VH	H	H
c_5	M	H	M	H	VH	VH	L	H	VH

C. Rating Requirements

The information is transformed to unify the heterogeneous information. Later fuzzy sets over S_t are transformed into linguistic 2-tuple.

In this example is applied a two-step aggregation process to compute a collective evaluation for software requirements. In our problem the 2-TOWAWA (4) is used to aggregate evaluations by requirement for each expert. In this case the weighting vectors are W=(0.3,0.2,0.2,0.15,0.15), V=(0.3,0.3,0.2,0.1,0.1) and H=0.3 giving more importance to the 2-TWA operator (70 %) in the aggregation (see Table II). These parameters could be established by means of the AHP [25] method and linguistic quantifiers [26].

Table II. An illustrative example of unified and aggregated information

	e_1			e_1			e_1		
	r_1	r_2	r_3	r_1	r_2	r_3	r_1	r_2	r_3
c_1	$(s_4,0)$	$(s_3,0)$	$(s_4,0)$	$(s_4,0)$	$(s_3,0)$	$(s_3,0)$	$(s_2,0)$	$(s_4,0)$	$(s_3,0)$
c_2	$(s_2,0)$	$(s_3,0)$	$(s_1,0)$	$(s_2,0)$	$(s_4,0)$	$(s_3,0)$	$(s_3,0)$	$(s_3,0)$	$s_2,0)$
c_3	$(s_4,-0.4)$	$(s_3,-0.2)$	$(s_2,0.4)$	$(s_4,-0.4)$	$(s_4,-0.4)$	$(s_2,0.4)$	$(s_2,0.4)$	$(s_4,-0.4)$	$(s_2,-0.4)$
c_4	$(s_3,0)$	$(s_2,0)$	$(s_3,0)$	$(s_4,0)$	$(s_1,0)$	$(s_4,0)$	$(s_4,0)$	$(s_2,0)$	$(s_3,0)$
c_5	$(s_2,0)$	$(s_3,0)$	$(s_2,0)$	$(s_3,0)$	$(s_4,0)$	$(s_4,0)$	$(s_1,0)$	$(s_3,0)$	$(s_4,0)$
2-TOWAWA	$(s_3,0.05)$	$(s_3,-0.15)$	$(s_3,-0.45)$	$(s_3, 0.3)$	$(s_3,0.32)$	$(s_3,0.179)$	$(s_3,-0,45)$	$(s_3,0.31)$	$(s_3,-0,35)$

To compute the collective for each requirement the 2-TWA operator is used with the weighting vector V=(0.5,0.2,0.3) (see Table III).

Table III. Collective evaluation for each requirement

r_1	(s_3,-0.0495)
r_2	(s_3,0.0795)
r_3	(s_3,.-0.2929)

Finally, we put in order all collective evaluations and we establish a ranking among requirements with the purpose of identifying the best ones. In the example the ranking is as follow: $r_2 \succ r_1 \succ r_3$.
After application in this illustrative example the model is found to be practical to use. The aggregation process gives a high flexibility so the model can be adapted to different situations. Interpretability of the linguistic output is another strength detected.

6. Conclusions

In this paper, we have proposed a prioritization model based on the decision analysis scheme that can manage different types of information (numerical and linguistic) and provide linguistic results in order to facilitate its understandability. We have applied the proposed model to an illustrative example. The model was found to be flexible and practical to use. The developing of software tool to automate the model is an area of future work.

7. Acknowledgment

This contribution has been supported by the research Project AGR-6487 and TIN-2009-08286.

8. References

[1] C. E. Otero, *et al.*, "A Quality-Based Requirement Prioritization Framework Using Binary Inputs," in *Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation*, 2010, pp. 187-192.

[2] N. R. Mead, "Requirements Prioritization Case Study Using AHP," Software Engineering Institute2008.

[3] A. Aurum and C. Wohlin, *Engineering and Managing Software Requirements*. New York: Springer, 2005.

[4] C. C. Wei, *et al.*, "An AHP-based approach to ERP system selection," *International Journal of Production Economics*, vol. 96, pp. 47-62, 2005.

[5] M. A. Iqbal, *et al.*, "A New Requirement Prioritization Model for Market Driven Products Using Analytical Hierarchical Process," in *International Conference on Data Storage and Data Engineering*, 2010.

[6] J. Azar, *et al.*, "Value-oriented requirements prioritization in a small development organization," *IEEE software*, pp. 32-37, 2007.

[7] K. Logue and K. McDaid, "Handling uncertainty in agile requirement prioritization and scheduling using statistical simulation," 2008, pp. 73-82.

[8] F. Herrera, *et al.*, "Managing non-homogeneous information in group decision making," *European Journal of Operational Research*, vol. 166, pp. 115-132, 2005.

[9] F. Herrera, *et al.*, "Computing with words in decision making: foundations, trends and prospects," *Fuzzy Optimization and Decision Making*, vol. 8, pp. 337-364, 2009.

[10] J. M. Merigó, "On the use of the OWA operator in the weighted average and its application in decision making," in *Proceedings of the World Congress on Engineering*, 2009.

[11] R. T. Clemen, *Making hard decisions: an introduction to decision analysis*: Duxbury Pres, 1996.

[12] A. Ejnioui, *et al.*, "Software requirement prioritization using fuzzy multi-attribute decision making," in *Open Systems (ICOS), 2012 IEEE Conference on*, 2012, pp. 1-6.

[13] M. Espinilla, *et al.*, "A heterogeneous evaluation model for assessing sustainable energy: A Belgian case study," in *Fuzzy Systems (FUZZ), 2010 IEEE International Conference on*, 2010, pp. 1-8.

[14] M. Ramzan, *et al.*, "Value based fuzzy requirement prioritization and its evaluation framework," in *Innovative Computing, Information and Control (ICICIC), 2009 Fourth International Conference on*, 2009, pp. 1464-1468.

[15] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, pp. 338-353, 1965.

[16] M. R. Berthold and D. J. Hand, *Intelligent Data Analysis: An Introduction*: Springer, 2010.

[17] M. S. Garcia-Cascales and M. T. Lamata, "Nueva aproximación al método tópsis difuso con etiquetas lingüísticas," presented at the ESTYLF 2010, Huelva, 2010.

[18] M. Espinilla Estévez, "Nuevos modelos de evaluación sensorial con información lingüística," DEA, Departamento de Informática, Universidad de Jaén, Jaen, 2009.

[19] G. Ruhe, "Software engineering decision support–a new paradigm for learning software organizations," in *Advances in Learning Software Organizations*, ed: Springer, 2003, pp. 104-113.

[20] M. Espinilla, *et al.*, "A comparative study of heterogeneous decision analysis approaches

applied to sustainable energy evaluation," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 20, pp. 159-174, 2012.

[21] F. Herrera and L. Martínez, "A 2-tuple fuzzy linguistic representation model for computing with words," *Fuzzy Systems, IEEE Transactions on*, vol. 8, pp. 746-752, 2000.

[22] J. Merigó, "New extensions to the OWA operators and its application in decision making," PhD Thesis, Department of Business Administration, University of Barcelona, 2008.

[23] J. M. Merigó, "A unified model between the weighted average and the induced OWA operator," *Expert Systems with Applications*, vol. 38, pp. 11560-11572, 2011.

[24] M. Espinilla, *et al.*, "A 360-degree performance appraisal model dealing with heterogeneous information and dependent criteria," *Information Sciences*, 2012.

[25] T. L. Saaty, *What is the analytic hierarchy process?*: Springer, 1988.

[26] V. Torra and Y. Narukawa, *Modeling decisions: information fusion and aggregation operators*: Springer, 2007.