NSS University of New Mexico

Neutrosophy for software requirement prioritization

Ronald Barriga Dias[1], Wilmer Ortiz Choez [2], Inelda Martillo Alcivar[3], Wilber Ortiz Aguilar[4]

[1]Universidad de Guayaquil, Facultad de Ciencias Matemáticas y Físicas, Guayaquil Ecuador. E-mail: ronald.barrigad@ug.edu.ec
[2]Universidad de Guayaquil, Facultad de Comunicaciòn Social, Guayaquil Ecuador. E-mail: wilmer.ortizc@ug.edu.ec
[3]Universidad de Guayaquil, Facultad de Ciencias Matemáticas y Físicas, Guayaquil Ecuador. E-mail: inelda.martilloa@ug.edu.ec
[4]Universidad de Guayaquil, Facultad de Ciencias Matemáticas y Físicas, Guayaquil Ecuador. E-mail: Wilber.ortiza@ug.edu

Abstract
Software engineers are involved in complex decisions that require multiples viewpoints. A specific case is the requirement prioritization process. This process is used to decide which software requirement to develop in certain release from a group of candidate requirements. Criteria involved in this process can involve indeterminacy. In this paper a software requirement prioritization model is develop based SVN numbers. Finally, an illustrative example is presented in order to show the proposed model.

Keywords: requirement engineering, software requirement prioritization, SVN numbers.

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 proposal present limitation for dealing with indeterminacy

In order to overcome the drawbacks identified, in this contribution we propose a novel requirement prioritization process based on SVN numbers.

In software requirement prioritization intervene different stakeholders approaching to the decision problem from a different points of view. It is moreover a multidimensional problems dealing with multiple criteria of diverse nature [8]. Therefore, the proposed model is based on a decision analysis scheme [9] 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 software prioritization. Section 3 shows the theory of neutrosophy. Section 4 presents our framework 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.

One frequent reason that causes low quality software is associated to problems related to identifying and selecting the most important requirements [10]. Software requirement prioritization can be modeled like a decision making problem, making it suitable to a decision analysis scheme[9]. Decision analysis is a discipline whose purpose is to help decision maker to reach a consistent decision [11].

Our proposal for a software requirement prioritization model dealing with indeterminacy is based on the classical decision analysis scheme. In this paper the software requirement prioritization process is modeled as a type of a Multi-Expert Multi-Criteria decision making problem due to the complexity of the problem where multiple criteria and experts are involved [10, 12].

In the software requirement prioritization process, it is very difficult to express reality in a quantitative way. Fuzzy set theory, introduced by Zadeh[13] 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 linguistic variable concept [14, 15]. This fact is important in software requirement prioritization where evaluation results are used to make decisions by software engineers in high complexity environment [16]. Current process of softeware prioritizationdon't deal with indeterminacy .

3. Neutrosophy

Neutrosophy [17] is a philosophy branch developed for dealing with indeterminacy (Figure 2). Neutrosophy have been the base for developing new methods to handle indeterminate and inconsistent information like neutrosophic sets an neutrosophic logic [18, 19] .

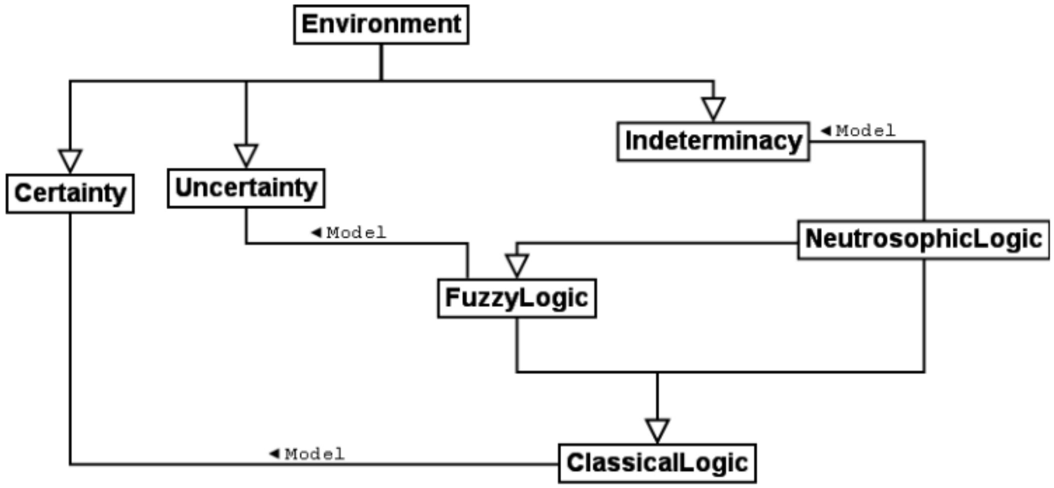

Fig. 1. Static context of Neutrosophic logic [20].

The truth value in neutrosophic set is as follows [21]:

Let N be a set defined as: $N = \{(T, I, F) : T, I, F \subseteq [0,1]\}$, a neutrosophic valuation n is a mapping from the set of propositional formulas to N , that is for each sentence p we have $v(\mathrm{p}) = (T, I, F)$.

Single valued neutrosophic set (SVNS) [22] was developed with the goal of facilitate the real applications of neutrosophic set and set-theoretic operators.

A single valued neutrosophic set (SVNS) has been defined as follows [22]:

Let X be a universe of discourse. A single valued neutrosophic set A over X is an object having the form :

$$A = \{\langle x, uA(x), rA(x), vA(x)\rangle : x \in X\} \quad (1)$$

where $u_A(x): X \to [0,1]$, $r_A(x),: X \to [0,1]$ and $v_A(x): X \to [0,1]$ with $0 \le u_A(x) + r_A(x) + v_A(x) :\le 3$ for all $x \in X$. The intervals $u_A(x)$, $r_A(x)$ y $v_A(x)$ denote the truth- membership degree, the indeterminacy-membership degree and the falsity membership degree of x to A, respectively.

Single valued neutrosophic numbers (SVN number) is denoted by $A=(a,b,c)$, where $a,b,c\in[0,1]$ and $a+b+c\le 3$.

4. A software requirement prioritization model

Our aim is develop a software requirement prioritization model based on the linguistic decision analysis schema that can deal with criteria evaluated with SVN numbers. The model consists of the following phases (graphically, Figure 2):

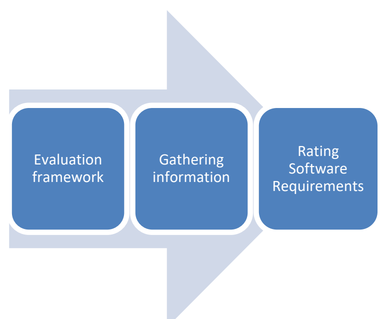

Figura 2. Scheme of the Model.

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

Each expert can use SVN numbers to asses each criteria, attending to its nature.

2. 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 [23] 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.

3. 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 more prioritized are identified. This phase in based the approach reviewed in the Section 3

A two-step aggregation process is developed with the aim of compute a global evaluation of each software requirement.

We obtain for each expert an assessment for each requirement.

The final aim of the rating process 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. In decision analysis schema aggregation operating are important for rating options. Some aggregation operators have been proposed for SVN numbers [17, 24]. Single valued neutrosophic weighted averaging (SVNWA) aggregation operator was proposed by Ye [24] for SVNSs as follows[25]:

$$F_w(A_1, A_2, \dots, A_n) = \left(1 - T_{A_j}(x)\right)^{w_j}, \langle 1 - \prod_{j=1}^{n} \prod_{j=1}^{n} \left(I_{A_j}(x)\right)^{w_j}, \rangle \prod_{j=1}^{n} \left(F_{A_j}(x)\right)^{w_j} \quad (2)$$

We propose this operator to establish different weights for each expert, taking into account their knowledge and their significance in software prioritization process

Rating of the requirements

The final step in the prioritization process 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.

For rating alternatives an ideal option is constructed [26, 27]. The evaluation criteria can be categorized into two categories, benefit and cost. Let C^+ be a collection of benefit criteria and C^- be a collection of cost criteria. The ideal alternative is defined as:

$$I = \left\{\left(max_{i=1}^{k} T_{U_j} | j \in C^+, min_{i=1}^{k} T_{U_j} | j \in C^-\right), \left(min_{i=1}^{k} I_{U_j} | j \in C^+, max_{i=1}^{k} I_{U_j} | j \in C^-\right), \left(min_{i=1}^{k} F_{U_j} | j \in C^+, max_{i=1}^{k} F_{U_j} | j \in C^-\right)\right\} = [v_1, v_2, \dots, v_n] \quad (4)$$

Alternatives are rating according Euclidean distance to I (2). Ranking is based in the global distance to the ideal. If alternative x_i is closer to I the distance measure (s_i closer) better is the alternative [28].

Alternatives could be rated according Euclidean distance in SVN [26, 29].

Let $A^* = (A_1^*, 3A_2^*, \dots, A_n^*)$ be a vector of n SVN numbers such that $A_j^* = (a_j^*, b_j^*, c_j^*)$ j=(1,2, … , n) and $B_i = (B_{i1}, B_{i2}, \dots, B_{im})$ (i = 1,2, … , m) be m vectors of n SVN numbers such that $B_{ij} = (a_{ij}, b_{ij}, c_{ij})$ (i = 1,2, … , m), (j = 1,2, … , n). Then the separation measure between $B_i's$ y A^* is defined as follows:

$$s_i = \left(\frac{1}{3}\sum_{j=1}^{n}\left\{\left(|a_{ij}-a_j^*|\right)^2 + \left(|b_{ij}-b_j^*|\right)^2 + \left(|c_{ij}-c_j^*|\right)^2\right\}\right)^{\frac{1}{2}}$$

$$(i = 1,2, \dots, m) \quad (2)$$

The best requirement is the one with the miimun distance to ideal.

5. Illustrative Example

In this section, we present an illustrative example in order to shown 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

The following linguistic terms are used (Table I).

Table I. Linguistic terms used to provide the assessments [26].

Linguistic terms	SVNSs
Extremely good (EG)	(1,0,0)
Very very good (VVG)	(0.9, 0.1, 0.1)
Very good (VG)	(0.8,0,15,0.20)
Good (G)	(0.70,0.25,0.30)
Medium good (MG)	(0.60,0.35,0.40)
Medium (M)	(0.50,0.50,0.50)
Medium bad (MB)	(0.40,0.65,0.60)
Bad (B)	(0.30,0.75,0.70)
Very bad (VB)	(0.20,0.85,0.80)
Very very bad (VVB)	(0.10,0.90,0.90)
Extremely bad (EB)	(0,1,1)

B. Gathering information

Once the evaluation framework has been determined the information about therequirements is gathered (see Table II).

Table II. 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	VVG	VG	EG	VVG	VG	G	M	VG	G
c_2	M	G	MB	M	VG	VG	G	M	MB
c_3	VG	M	M	VG	VVG	M	MB	G	B
c_4	G	M	VG	VG	B	VG	VG	G	G
c_5	M	G	M	G	VG	VVG	B	G	VG

C. Rating Requirements

In this example, is applied a two-step aggregation process to compute a collective evaluation for software requirements. In our problem the SVNWA is used to aggregate evaluations by requirement for each expert. In this case the weighting vectors to compute the collective evaluation is V=(0.3,0.3,0.4) .

Table III. An illustrative example of unified and aggregated information

	r_1	r_2	r_3
c_1	(0.24, 0.2, 0.12)	(0.18, 0.18, 0.14)	(0.19, 0.0, 0.0)
c_2	(0.41, 0.44, 0.35)	(0.32, 0.3, 0.25)	(0.46, 0.44, 0.35)
c_3	(0.38, 0.0, 0.17)	(0.29, 0.27, 0.19)	(0.54, 0.61, 0.5)
c_4	(0.21, 0.21, 0.17)	(0.49, 0.49, 0.41)	(0.21, 0.21, 0.17)
c_5	(0.49, 0.49, 0.41)	(0.24, 0.25, 0.2)	(0.26, 0.23, 0.16)

From this information, the ideal alternative is calculated (Table IV).

Table IV. Ideal alternative

	E^+
c_1	(0.2,0,0)
c_2	(0.4,0.3,0.25)
c_3	(0.38, 0.61,0.5)
c_4	(0.49,0.21,0.17)
c_5	(0.24,0.49,0.41)

The results of the calculation of the distances allow requeriment.

Table V. Distance to ideal alternative

r_1	0.21
r_2	0.38
r_3	0.45

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 case study the ranking is as follow: $r_1 \succ r_2 \succ r_3$

After application in this case study 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.

6. Conclusions

In this paper, we have proposed a prioritization model based on the decision analysis scheme that can manage SVN numbers. 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.

References

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

2 Mead, N.R., Requirements Prioritization Case Study Using AHP. 2008, Software Engineering Institute.

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

4 Wei, C.C., C.F. Chien, and M.J.J. Wang, An AHP-based approach to ERP system selection. International Journal of Production Economics, 2005. **96**(1): p. 47-62.

5 Iqbal, M.A., A.M. Zaidi, and S. Murtaza. A New Requirement Prioritization Model for Market Driven Products Using Analytical Hierarchical Process. in International Conference on Data Storage and Data Engineering. 2010. IEEE.

6 Azar, J., R.K. Smith, and D. Cordes, Value-oriented requirements prioritization in a small development organization. IEEE software, 2007: p. 32-37.

7 Logue, K. and K. McDaid. Handling uncertainty in agile requirement prioritization and scheduling using statistical simulation. 2008. IEEE.

8 Herrera-Viedma, E., et al., Applying Linguistic OWA Operators in Consensus Models under Unbalanced Linguistic Information Recent Developments in the Ordered Weighted Averaging Operators: Theory and Practice, R. Yager, J. Kacprzyk, and G. Beliakov, Editors. 2011, Springer Berlin / Heidelberg. p. 167-186.

9 Clemen, R.T., Making hard decisions: an introduction to decision analysis. 1996: Duxbury Pres.

10 Ejnioui, A., C.E. Otero, and A.A. Qureshi. Software requirement prioritization using fuzzy multi-attribute decision making. in Open Systems (ICOS), 2012 IEEE Conference on. 2012. IEEE.

11 Espinilla, M., et al. A heterogeneous evaluation model for assessing sustainable energy: A Belgian case study. in Fuzzy Systems (FUZZ), 2010 IEEE International Conference on. 2010. IEEE.

12 Ramzan, M., et al. Value based fuzzy requirement prioritization and its evaluation framework. in Innovative Computing, Information and Control (ICICIC), 2009 Fourth International Conference on. 2009. IEEE.

13 Zadeh, L.A., Fuzzy sets. Information and Control, 1965. **8**(3): p. 338-353.

14 Berthold, M.R. and D.J. Hand, Intelligent Data Analysis: An Introduction. 2010: Springer.

15 Garcia-Cascales, M.S. and M.T. Lamata, Nueva aproximación al método tópsis difuso con etiquetas lingüísticas, in ESTYLF 2010. 2010: Huelva.

16 Ruhe, G., Software engineering decision support–a new paradigm for learning software organizations, in Advances in Learning Software Organizations. 2003, Springer. p. 104-113.

17 Smarandache, F., A Unifying Field in Logics: Neutrosophic Logic. Philosophy, 1999: p. 1-141.

18 Smarandache, F., A Unifying Field in Logics: Neutrosophic Logic. Neutrosophy, Neutrosophic Set, Neutrosophic Probability: Neutrsophic Logic. Neutrosophy, Neutrosophic Set, Neutrosophic Probability. 2005: Infinite Study.

19 Vera, M., et al., Las habilidades del marketing como determinantes que sustentaran la competitividad de la Industria del arroz en el cantón Yaguachi. Aplicación de los números SVN a la priorización de estrategias. Neutrosophic Sets & Systems, 2016. **13**.

20 Salmeron, J.L. and F. Smarandache, Redesigning Decision Matrix Method with an indeterminacy-based inference process. Multispace and Multistructure. Neutrosophic Transdisciplinarity (100 Collected Papers of Sciences), 2010. **4**: p. 151.

21 Rivieccio, U., Neutrosophic logics: Prospects and problems. Fuzzy sets and systems, 2008. **159**(14): p. 1860-1868.

22 Wang, H., Smarandache, F. et al., Single valued neutrosophic sets. Review of the Air Force Academy, 2010(1): p. 10.

23 Espinilla, M., et al., A 360-degree performance appraisal model dealing with heterogeneous information and dependent criteria. Information Sciences, 2012.

24 Ye, J., A multicriteria decision-making method using aggregation operators for simplified neutrosophic sets. Journal of Intelligent & Fuzzy Systems, 2014. **26**(5): p. 2459-2466.

25 Biswas, P., S. Pramanik, and B.C. Giri, TOPSIS method for multi-attribute group decision-making under single-valued neutrosophic environment. Neural computing and Applications, 2016. **27**(3): p. 727-737.

26 Şahin, R. and M. Yiğider, A Multi-criteria neutrosophic group decision making metod based TOPSIS for supplier selection. arXiv preprint arXiv:1412.5077, 2014.

27 Leyva-Vázquez, M., K. Pérez-Teruel, and R.I. John. A model for enterprise architecture scenario analysis

based on fuzzy cognitive maps and OWA operators. in Electronics, Communications and Computers (CONIELECOMP), 2014 International Conference on. 2014. IEEE.

28 Leyva-Vázquez, M., et al., Técnicas para la representación del conocimiento causal: un estudio de caso en Informática Médica. Revista Cubana de información en ciencias de la salud, 2013. **24**(1): p. 73-83.

29 Ye, J., Single-valued neutrosophic minimum spanning tree and its clustering method. Journal of intelligent Systems, 2014. **23**(3): p. 311-324.

30 Abdel-Basset, M., Mohamed, M., & Sangaiah, A. K. (2017). Neutrosophic AHP-Delphi Group decision making model based on trapezoidal neutrosophic numbers. Journal of Ambient Intelligence and Humanized Computing, 1-17.https://doi.org/10.1007/s12652-017-0548-7

31 Abdel-Basset, M., Mohamed, M., Hussien, A. N., & Sangaiah, A. K. (2017). A novel group decision-making model based on triangular neutrosophic numbers. Soft Computing, 1-15. http://doi.org/10.1007/s00500-017-2758-5

32 F. Smarandache, M. Ali, Neutrosophic Triplet as extension of Matter Plasma, Unmatter Plasma, and Antimatter Plasma, 69th Annual Gaseous Electronics Conference, Bochum, Germany, Veranstaltungszentrum & Audimax, Ruhr-Universitat, October 10-14, 2016, http://meetings.aps.org/Meeting/GEC16/Session/HT6.112

33 F. Smarandache, Neutrosophic Perspectives: Triplets, Duplets, Multisets, Hybrid Operators, Modal Logic, Hedge Algebras. And Applications. Pons Editions, Bruxelles, 325 p., 2017.

34 Topal, S. and Öner, T. Contrastings on Textual Entailmentness and Algorithms of Syllogistic Logics, Journal of Mathematical Sciences, Volume 2, Number 4, April 2015, pp 185-189.

35 Topal S. An Object- Oriented Approach to Counter-Model Constructions in A Fragment of Natural Language, BEU Journal of Science, 4(2), 103-111, 2015.

36 Topal, S. A Syllogistic Fragment of English with Ditransitive Verbs in Formal Semantics, Journal of Logic, Mathematics and Linguistics in Applied Sciences, Vol 1, No 1, 2016.

37 Topal, S. and Smaradache, F. A Lattice-Theoretic Look: A Negated Approach to Adjectival (Intersective, Neutrosophic and Private) Phrases. The 2017 IEEE International Conference on INnovations in Intelligent SysTems and Applications (INISTA 2017); (accepted for publication).

38 Taş, F. and Topal, S. Bezier Curve Modeling for Neutrosophic Data Problem. Neutrosophic Sets and Systems, Vol 16, pp. 3-5, 2017.

Received: August 10, 2017. Accepted: August 31, 2017.