

Prioritization of Software Requirements: a Cognitive Approach

Martin Azzolini¹, Lucia Isabel Passoni¹

¹Universidad Nacional de Mar del Plata, Mar del Plata, Argentina

martinazzolini@hotmail.com, isabel.passoni@gmail.com

Abstract

This approach provides a novel methodology for addressing a well-known problem of Software Engineering, the prioritization of software requirements. The proposal combines the use of tools such as the Analytic Hierarchy Process (AHP), the Self-Organizing Maps (SOM) and also methods belonging to the Cognitive Psychology to build a Conceptual Framework to assess the knowledge level that stakeholders have of the domain of the software application. Hence the cognitive paradigm helps to interpret the stakeholders clusters, built up with their preferences on the prioritization of the software requirements.

Keywords: Software Engineering, Prioritization of Software Requirements, Cognitive approach; AHP, SOM.

1. Introduction

The Software Requirements Prioritization involves the formulation of a decision process of a complex nature. In Hadad *et al.*, [1] provides that to assign priorities in the requirements leads to discern issues inherent dependency thereof, the multiplicity of interests among the stakeholders, resource constraints, business requirements, implementation costs, including other factors. Consequently, the prioritization of requirements has been recognized as one of the most important activities within the software development process [2-4].

The Prioritization of Software Requirements can be defined as the "process that determines the order of implementation of the requirements" or also as "the process that determines the order of importance of requirements based on the perception of stakeholders" [5]. Given this definition, it is assumed that stakeholders valued a particular requirement based on the importance suggests its implementation.

Hence, in this paper we propose a methodology of requirements prioritization that deals with the existence of dissimilar preferences among the stakeholder population, under a cognitive framework approach that would explain this variability of preferences.

Nuseibeh and Easterbrook [6] determined that in the context of software system implementation, the Requirements Engineering (RE) should provide a conceptual framework which would explain how stakeholders relate

to each other, how they perceive and understand the surrounding reality, and how their actions affect organizational sociology. Consequently, this framework must be addressed like an important philosophical component within the conceptual structure of the Requirements Engineering (RE). That is, should provide a methodological framework applicable to the interpretation and understanding of the terminology used by stakeholders, their views and objectives.

As a result, RE draws on the Cognitive and Social Sciences provided the effects of a body of common knowledge, designed to satisfy the concerns that underlie human interaction in the context of implementation of the basic processes of RE: Elicitation, Specification and Validation of Requirements Loucopoulos and Karakostas [7]. Meanwhile, Cognitive Psychology provides a conceptual framework, to deal with the descriptions of the stakeholders requirements, Posner [8].

As originally proposed in [6], experts in the problem domain, usually possess a wealth of tacit knowledge rather hard to communicate, so in many situations the users' replies to the concerns raised by the Requirements Engineers do not match with their actual needs. Hence, in Burg [9] was considered the use of Linguistic Techniques for analyzing the pattern of organizational communication during the execution of the elicitation process of software requirements.

In a design context where there are different profiles of stakeholders, the prioritization of requirements is a complex task. The negotiating approaches must be supported by tools that allow at first, the grouping of stakeholders according to their priorities and secondly its validation based on the knowledge of the application domain. So, clusters of stakeholders can be discovered, given the existence of similarities in their preferences. Furthermore, as shown in Martinez Carod [10], during the elicitation process emerge the fact that stakeholders have different expectations for the system to be developed. In fact, they are legitimized by the diverse interests of parties involved, their backgrounds and needs.

Thus, personal factors determine the perception of stakeholders, biased by different interpretations about the requirements of the future system. Based on this perspective; we propose the convenience of describing and analyzing the concept of semantic memory. Semantic memory is referred to the memory of meanings, under-

standings, and other concept-based knowledge that is not related to personal experiences, Vivas [11]. Hence, the meaning of a concept seems to emerge directly of its intrinsic properties. Instead, the meaning attributed to an entity, phenomenon or process seems to emerge from the relative weight of relations that the subject has established in its history with the reference object [12-14].

In this point, we consider that the domain of a software application can be addressed as a Conceptual Structure, as a picture of the stakeholders' semantic memory. Then, the requirements engineering process must systematize the analysis of this conceptual structure, in order to provide a System Requirements Specification (SRS). The prioritization of these specifications shall reflect the level of the domain knowledge evidenced by stakeholders as a result of the analysis of this Conceptual Structure.

Hence, in this paper, we propose to detect clusters of stakeholders preferences revealed via the Analytic Hierarchy Process (AHP), Saaty [15-17]. The cluster detection is carried out using Self Organizing Maps (SOM), Kohonen [18-20], as a tool that allows the interpretation of how the preference vectors impact in the cluster generation. Then, a design negotiations methodology is addressed focused on the cognitive performance of each of these groups. In this regard, stresses the idea developed in Damian et al [21] which suggests that the process of negotiation of requirements will be more effective as the stakeholders share a common understanding of the domain, and also promote communication whose content is semantically valid accepted by the different parties involved.

An efficient clustering of stakeholders according to their preferences, followed by the assessment of the cognitive characteristics within the groups, is essential to prioritize requirements when the stakeholders profiles exhibits diversity. Because of the need of interpretation of the clustering process, we propose the use of self organizing maps to perform the mapping of the preferences vectors of the stakeholders. The importance of this process is closely related with the assertion that one of the critical factors in the success of a Software Project, consists precisely in achieving the requirements prioritization in a relatively early stage of software life cycle [22-23].

2. Methodology Introduction

In order to relieve the preferences of the stakeholders the Analytic Hierarchy Process method (AHP) of Saaty will be used [15-17], under the reformulation proposed by Karlsson [24]. For this, the first step is to formulate a matrix $n \times n$, where n is the number of system requirements to be evaluated by the stakeholders:

$$A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \dots & \dots & \dots \\ a_{n1} & \dots & a_{nn} \end{pmatrix} \quad (1)$$

In (1), each term of cell, implies a comparison-pairs between the Requirement of order i and its consecutive (a_{i+i}) by each stakeholder. To carry out the comparisons, using scale ratios: 1 to 9. The objective is to measure the intensity or relative importance of one requirement over another, according to the preference for a certain stakeholders surveyed (1: Both requirements are also preferred, 9: Requirement is preferred with the highest degree of intensity possible compared to the other).

Also, all the diagonal terms of (1) are 1, because, comparing two requirements are the same. Collected preference matrix, the third step is to normalize each cell by weighting the sum of the terms that make up the reference column. I.e., dividing each $a_{i,j}$ by the respective sum of the cell column. Finally, the fourth step will be to estimate the vector of self-value weighted by the number of requirements to prioritize, give the preference vector of stakeholders:

$$w = \frac{1}{n} \cdot \begin{pmatrix} w_{11} \\ \dots \\ w_{nn} \end{pmatrix} \quad (2)$$

According to (2), Saaty [15] proposes an inconsistency index (RI) of the preference vector revealed by the stakeholders. When $RI > 0.1$, then should not consider the vector provided by the stakeholders. Finally, considering that each vector of preference (2) satisfies the condition $RI < 0,1$, will use a SOM network for preference clustering. While clustering algorithms, in general, attempt to partition data into natural groups by maximizing inter-cluster distance and minimizing intra-cluster distance, the SOM performs a clustering of a slightly different nature. The SOM can be thought of as a spatially constrained form of k -means clustering or as a projection maintaining the neighborhood relations in the data.

The two-level clustering methodology (clustering of the SOM rather than clustering the data directly) is preferred because of the reduction of the computational cost. Even with relatively small number of samples, many clustering algorithms—especially hierarchical ones—become intractable. For this reason, it is convenient to cluster a set of prototypes rather than directly the data.

Another benefit is noise reduction. The prototypes are local averages of the data and, therefore, less sensitive to random variations than the original data.

To train the SOM, let $X_i \in R^p$, $i = 1 \dots n$, p one-dimensional data set, and $V_j \in R^p$ $j = 1 \dots c$, a set of vectors of dictionaries, the Kohonen update rule is as follows: (3)

$$V_{j,t} = V_{j,t-1} + \alpha_t h_{r,t} (X_i - V_{j,t-1}) \quad (3)$$

Where α_t represents the learning factor, which is defined as a decreasing function that controls the magnitude of changes in each iteration t , which begins taking maximum value 1 and will decrease up to 0.05. Also $h_{r,t}$ is the function that controls the size of the vicinity of the nodes (r) to be updated during the training process and will decrease monotonically until reaching the minimum (one

cell). To provide stakeholders groups, a cluster methodology is applied to SOM codebooks, like k-means. The Davies-Bouldin index is used in order to evaluate the quality of the obtained clusters [25].

Each group will be represented by a list of requirements generated by the average of the preference of each requirement within the cluster. Hence, each group shows a preference vector ordered from highest to lowest with the requirements according with the average of their preferences (SOM's components variables) belonging to its cluster.

In order to assess the knowledge level that stakeholders have of the domain of the software application, they will be asked to identify those that are considered key concepts. After all the surveyed stakeholders voted on the list of candidate concepts (proposed by the Requirements Engineer), a subset of key concepts is built with those that obtained at least 51% of the votes. These concepts will be understood as the Conceptual Structure.

Since a concept may be expressed or associated by a set of words [26-27], we will ask each of the stakeholders to identify at least ten words that define the key concepts of the domain. That is, each of the key concepts will have associated a list of words identified by each of the stakeholders. Then, the defining words lists are processed by the algorithm so called "Definition Finder", Vivas *et al.*, [27]. The method assigns a relative weight to each of the words depending on the frequency of occurrence and the relative order assigned by each of the subjects that compose the aforementioned whole.

After application of Definition Finder, each concept is composed of a succession of relative weights that specify the order of importance of each of the defining words that make up the concept. Then, with the aim of evaluating the domain of each of the key concepts for each group of the stakeholders, the *VTV* algorithm is used [28]:

$$VTV_c(k) = \frac{\beta_0}{\beta_1} \left[e^{\beta_1 n} - e^{\beta_1} \right] \quad (4)$$

Wherein (4):

$VTV_c(k)$: refers to Value Total Valuations of analyzed concept *c* in group *k* already discovered by SOM.

β_0 : refers to the weight acquired by the "semantic relevance" of concept analysis.

β_1 : refers to the distinctive effect that takes the concept analyzed.

n: refers to the amount of defining socially identified in the concept discussed.

VTV method, calculates the sequence of the relative weights of each of the key concepts of the domain (Definition Finder). Therefore, it is considered as an estimator of the semantic field of each of the concepts. Each group of stakeholders assigned a value of VTV_c for each of the concepts. Getting an average of *VTV* in each group over the whole concepts $\overline{VTV}_{(k)}$, we can approach the level of knowledge that this group has been able to get around the key concepts that summarize the uses and implications of the software tool. Finally the prioritized requirement list

is generated according to a weighted average of preferences values with the $\overline{VTV}_{(k)}$, of each stakeholder group:

$$VMP_{UR(i),r} = \sum_{i=1}^q \sum_{k=1}^p \left(\frac{\overline{VTV}_k}{VTV_{accumulate}} \right) \cdot VMP_{R(i,k)} \quad (5)$$

Where $k=1..p$ (p = groups)

$i=1..q$ (q =requirements)

Where $[VMP_{UR(i),r}]$ is the set of priority values of the Requirements, computed with the preference vectors of each group $[VMP_{R(i,k)}]$, weighted with their degree of knowledge of the key concepts of the domain. Hence the achieved prioritization of requirements will depend on the degree of knowledge that stakeholders have in terms of the key concepts that synthesize the uses and implications of a software tool (Conceptual Structure).

Since the method for prioritizing requirements (5) considers the average value of *VTV*, as a mechanism to weight the preferences of each group, it is necessary evaluate whether each mean is significantly different from the rest, if not, this weighting scheme would be redundant. To perform this task the non-parametric Kruskal-Wallis test [29], is used, with the null hypothesis (H_0) of *VTV* equal population, versus the alter-native hypothesis (H_1) that at least one of them differs from the rest. The hypothesis shall be tested on the basis of a 5 % significance level.

3. Case Study

A case of a Web Platform must support the functionality of a Workflow. This particular tool is composed of 11 Functional Requirements can be summarized in 4 User Requirements. The experiment to be performed is to elicit the preference of 50 stakeholders, and then, define the order of implementation of the User Requirements.

The experiment in question is complex given the variety of existing stakeholders and their spatial distribution. That is, not all stakeholders living in the same geographical space, however, all share the profession of researcher, but they are employed in various scientific disciplines.

4. Results

To process different semantic productions of the key concepts of the domain (defining words), we use the Software Version 0.93 Definition Finder developed jointly by the GIA (Grupo de Inteligencia Artificial, Facultad de Ingenieria de la Universidad Nacional de Mar del Plata) and the CIMEPB (Centro de Investigación en Procesos Básicos, Metodología y Educación). To clustering of the preferences of stakeholders, we apply version 7.8.0 (R2009a) of the MATLAB software with the library "SOM Toolbox" developed in Alhoiemi, et al [30]. Last, we use the SPSS version 17.0 to run the algorithm of Kruskal Wallis and detect significant differences between the averages of *VTV*.

The stakeholders have considered the existence of 4 key concepts that synthesize the uses and implications of the workflow tool. That is, the concepts "Soft-ware Tools", "Research Environment", "Collaborative Workflow" and "Interdisciplinary Insights", meet the 51.35% of the votes of the stakeholders. These concepts are considered as the Conceptual Structure.

In Table 1 we see the list of preference vectors for each of the stakeholders.

A SOM network was trained using open source tool provided by Alhoiemiet et al [30]. The analysis of the obtained SOM unified suggests the existence of three well defined clusters of stakeholders, according with their Users Requirements preferences, extracted from the AHP method.

Preference Vectors (Relieved by AHP Method)				
Stakeholders	UR ₁	UR ₂	UR ₃	UR ₄
S ₁	0,50	0,13	0,14	0,22
S ₂	0,15	0,21	0,23	0,42
S ₃	0,39	0,09	0,15	0,37
...
S ₅₀	0,42	0,06	0,18	0,33

Table 1 Preference vectors released by the AHP method.

To evaluate the Som's clusters quality the Davies-Bouldin index (Fig. 1) sets a minimum value of 0.45 for the given configuration of three groups of stakeholders found with the k-means algorithm.

Based on the groups provided by the SOM network, the preferences average values were computed in each group. The aggregation has been linear, as is stated in Karlsson [24].

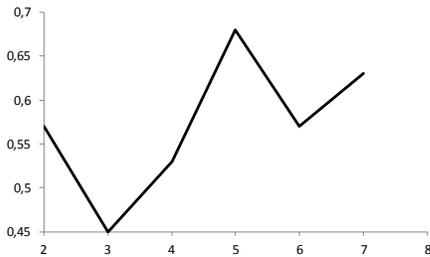


Fig. 1 Davies-Bouldin index of the proposed SOM clustering

As shown in Table 2, Groups 1 and 2 show the same pattern of priority. That is, their User Requirements were ranked in the following sequence: 1,4,3,2. However, the Group 3 provided another order: 4, 3, 2, 1.

Based on the groups provided by the SOM network, the preferences average values were computed in each group. The aggregation has been linear, as is stated in Karlsson [24]. As shown in Table 1, Groups 1 and 2 show the same pattern of priority. That is, their User Requirements were ranked in the following sequence: 1,4,3,2. However, the Group 3 provided another order: 4, 3, 2, 1.

Average values of preference by group				
Groups	UR ₁	UR ₂	UR ₃	UR ₄
1	0,57	0,06	0,13	0,24
2	0,43	0,07	0,15	0,35
3	0,16	0,22	0,25	0,37

Table 2 Average values of preferences by group

Then, the semantic field VTV for the Conceptual Structure into each group was computed, as shown in Table 3.

Concept	Group 1	Group 2	Group 3
Software Tool	4,59	4,61	3,34
Research Environment	4,53	4,17	2,44
Collaborative Work	4,47	2,63	2,42
Interdisciplinary Insights	4,91	3,02	2,43
Average	4,63	3,61	2,66

Table 3 Semantic field of the key concepts of the Dominion, by group

In hindsight, a Kruskal-Wallis test was applied to the vector whose components are group averages VTV of each concept in each group. The test result confirms the rejection of the null hypothesis ($p\text{-value} = 0.03$). This implies that at least one group has shown a different degree of knowledge about the key concepts of the do-main. Finally, we apply (5) to obtain the priorities of user requirements list ordered values from highest to lowest, as observed in (6).

$$VMP_{UR(1,2,3,4)Total} = \begin{bmatrix} 1^{\circ}UR_1 (42\%) \\ 2^{\circ}UR_4 (31\%) \\ 3^{\circ}UR_3 (17\%) \\ 4^{\circ}UR_2 (10\%) \end{bmatrix} \quad (6)$$

Consequently, this result supports the fact that the method of prioritizing high priority be given to the preference of those groups that have a higher level of domain knowledge, as measured by its average VTV.

5. Conclusions

In this paper, we proposed a method to prioritize requirements through the aggregation of dissimilar preferences elicited in a population of stakeholders. We state that the preference of stakeholders is functional to their semantic memory, evidenced by a conceptual structure that summarizes the uses and implications of future software tool (Key Concepts).

The proposed methodology discovered stakeholders groups according to their preferences on the prioritization requirements (AHP-SOM). These clusters showed high correspondence with the stakeholder's knowledge level

on the key concepts of the domain. Finally, the ranked prioritization list has allowed adding the preferences expressed by each group, depending on their respective cognitive performance (VTV).

This approach provides an appropriate methodology for addressing a complex problem such as the prioritization of software requirements, using elements of disciplines that are not usual to address in the field of Software Engineering. We have developed some field experiments that showed the success of combining tools from the Computational Intelligence area, specifically Neural Networks (Self-Organized Maps) and also from Cognitive Psychology, like a Conceptual Structure that shows a picture of the semantic memory of the dominion.

6. Future Works

From this experience, where the design of a prioritization model of system requirements from the segmentation of the stakeholder's preferences was pointed, we will continue to consider how to include their cognitive profile to improve the clusters segmentation. We will try to develop a method of aggregation of preference inter-groups that will weight cognitive metrics in order to obtain a global preferences vector. Given the existence of groups of stakeholders whose preferences can differ from the overall set up, we will propose the design of a negotiation process, in order to reconcile the differences explained by their different cognitive approach.

7. References

- [1] Hadad GDS, Doorn JH, Ridao M, Kaplan GN (2009) Facilitando la Asignación de Prioridades a los Requisitos, *WER2009 12th Workshop on Requirements Engineering*, ISBN: 978-956-319-941-3, Universidad Técnica Federico Santa María, Valparaíso, Chile, pp.75-84
- [2] Gomes, A; Pettersson, A (2007) "Market Driven Requirements Engineering Process Model –MDREP". Karlskrona: Blekinge Institute of Technology.
- [3] Lehtola, L; Kauppinen, M. (2006) "Suitability of Requirements Prioritizations Methods for Market Driven Software Product Development". *Software Process: Improvement and Practice*. 11. (1). Pg: 7-19.
- [4] Ngo-The, A; Ruhe, G (2005) "Decision support in requirement engineering". In *Aurum & Wohlin (Eds). Engineering and managing software requirements*. Pg: 267-286. Springer Berlin Heidelberg.
- [5] Firesmith, D (2004) "Prioritizing Requirements". *Journal of Object Technology*. 3. (8). Pg: 35-47.
- [6] Nuseibeh, B; Easterbrook, S (2000) "Requirements Engineering: A Roadmap". *Proc. 22nd Int'l Conf. Software Eng.*, pp. 35-46.
- [7] Loucopoulos, P; Karakostas, V (1995). *System Requirement Engineering*. 1era. Ed. McGraw. New York. USA.
- [8] Posner, M (1993). "Foundations of Cognitive Science". MIT Press.
- [9] Burg, J (1997) "Linguistic Instruments in Requirements Engineering". Amsterdam : IOS Press.
- [10] Martinez Carod, N (2007). "Gestión de Preferencias de Requerimientos basadas en Técnicas Cognitivas". Departamento de Ciencias de la Computación. UNComa. Argentina.
- [11] Vivas, J. (2009). "Modelos de Memoria Semántica". En J. Vivas (Comp.). *Evaluación de redes semánticas. Instrumentos y Aplicaciones*". Mar del Plata: Eudem. ISBN 978-987-05-5903-0 (En Prensa).
- [12] Murphy, G. L. (2002). "The big book of concepts". Massachussets: MIT Press.
- [13] Rogers, T. T. y McClelland, J. J. (2004) *Semantic cognition: A parallel distributed approach*. Massachussets: MIT Press.
- [14] McNamara, T (2005) "Semantic priming: Perspectives from memory and word recognition". Nueva York: Psychology Press.
- [15] Saaty, T (1980). *The Analytic Hierarchy Process*. NY. McGraw- Hill.
- [16] Saaty, T,L. (1986). "Axiomatic foundations of the analytic hierarchy process". *Management Science*. 32. Pp: 841-855.
- [17] Saaty, T,L. (2001) "Decision-Making whit the AHP: Why is the principal eigenvector necessary?". ISAHP 2001, Berne, Switzerland,
- [18] Kohonen, T. (1982) Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43. Pg: 59-69.
- [19] Kohonen, T(1990). "Self-Organizing Maps" *Proceedings of the IEEE*. 78.
- [20] Kohonen, T(2001). *Self-Organizing Maps*. 30. Springer Verlag.
- [21] Damian, D; Lanubile, F; Mallardo, T (2008). "On the need for mixed media in distributed requirement negotiations". *IEEE Transactions on Software of Engineering*. Vol: 34, 1, January.
- [22] In, H., Boehm, B., Rodgers, T., and Deutsch, M (2001) "Applying WinWin to Quality Requirements: A Case Study", *IEEE International Conference on Software Engineering (ICSE 2001)*, IEEE Computer Society Press, Toronto, Canada, May 12-19, 2001, pp. 555- 564.
- [23] Boehm, B. and In, H. (1996) "Identifying Quality-Requirement Conflicts", *IEEE Software*, Vol. 13, No. 2, March.
- [24] Karlsson, J. (1996). "Software Requirements Prioritizing". *Proceedings of ICRE '96*. IEEE Software.
- [25] Davies, D; Bouldin, D (1979) "A cluster separation measure," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-1, pp. 224–227, Apr.
- [26] Vivas, J., Huapaya, C., Lizarralde, F., Arona, G., Comesaña, A. (2009). "La evaluación semántica en entornos de nuevas tecnologías de la información y la comunicación". En M. Concepción Rodríguez y V. Padilla Montemayor (Comp.) *Interacción y medición*

en ambientes de aprendizaje electrónicamente mediados. Monterrey: Ed. Trillas. 92-111.

- [27] Vivas, J.; Ledesma, R. (2009). "Evaluación de la memoria semántica. El método Distsem". En Evaluación de redes semánticas. Instrumentos y Aplicaciones. MDP: Eudem. ISBN 978-987-05-5903-0.
- [28] Vivas, J; Azzolini, M; Vivas, L (2010) "Buscador de Definidoras y VTV. Utilidad de algunas técnicas de generación de normas para la producción de atributos semánticos y la definición del campo semántico. VII Congreso Iberoamericano de Psicología. Oviedo. 22 de Julio de 2010.
- [29] Kruskal, W; Wallis, W (1952) "Allen Wallis. Use of ranks in one-criterion variance analysis" *Journal of the American Statistical Association* 47 (260): 583–621.
- [30] Alhoiemi E; Himberg J; Parhankangas J; Vesanto J (2005) "SOM Toolbox". Laboratory of Computer and Information Science. SOM Toolbox Team. Finland.