Clustering Stakeholders for Requirements Decision-Making

Varsha Veerappa[1], Emmanuel Letier[1]

[1] University College London, Gower Street, London WC1E 6BT, United Kingdom
{v.veerappa, e.letier} @cs.ucl.ac.uk

Abstract. [Context and motivation] Novel web-based requirements elicitation tools offer the possibility to collect requirements preferences from large number of stakeholders. Such tools have the potential to provide useful data for requirements prioritization and selection. **[Question/problem]** However, existing requirements prioritization and selection techniques do not work in this context because they assume requirements ratings from a small number of stakeholders' groups, rather than from a large number of individuals. They also assume that the relevant groups of stakeholders have been identified a priori, and that all stakeholders within a group have the same preferences. **[Principal ideas/results]** This paper aims at addressing these problems by applying cluster analysis techniques used in the area of market segmentation for identifying relevant groups of stakeholders to be used for requirements decision-making. **[Contribution]** We describe clustering analysis techniques that can be used in this context and evaluate their adequacy on a pilot case study.

Keywords: Stakeholder segmentation, cluster analysis, web-based requirements elicitation, requirements prioritization and selection.

1 Introduction

There is an increasing trend towards using web-based application such as forums, wikis, and recommender systems to elicit and prioritizing requirements from very large number of stakeholders [1], [2]. For example, StakeSource is a web-based toolset that helps requirements engineers identify project stakeholders, elicit product requirements and stakeholders' preferences for these requirements by asking stakeholders to recommend other stakeholders, propose new requirements, and rate already submitted requirements [3]. These systems help collecting data that could be used for understanding stakeholders and their preferences, identifying conflicts, and guiding requirements selection and prioritization.

There exists a wide range of qualitative and quantitative techniques for identifying the best tradeoffs among the preferences of multiple stakeholders [4]. Cost-value based requirements prioritization techniques rely on eliciting the relative costs and value of each requirements for each stakeholders group [5]. By assigning weights to all stakeholders groups, one can compute the overall value of a requirement as the weighted sum of its value for each stakeholders group, and rank the set of requirements accordingly. Different variants of this approach are used in practice [6], [7], [8]. Generating a full ranking of requirements based on a single numerical value hides conflicts between stakeholders instead of exposing them. More recent requirements selection techniques have therefore looked at the problem as a multi-criteria decision problems and developed support for exploring the space of optimal solutions and reasoning about the fairness of the requirements selection [9], [10], [11].

All these techniques have been developed in a context where requirements values are elicited for a small number of stakeholders' groups only. They do not scale to the context of online requirements elicitation tools where values are elicited from a large number of *individual* stakeholders. Furthermore, they assume that homogenous groups of stakeholders can be identified a priori, and that the value of a requirement for a given group represents the consensus value for all stakeholders within that group (which leaves the problem of handling divergences of opinion within a group unresolved). An additional difficulty specific to online elicitation tools is that some groups of stakeholders are likely to be under-represented or over-represented in the collected ratings. For example, stakeholders who have more time to express their preferences online are likely to be over-represented compared to more busy stakeholders whose opinion may be no less important to the project success.

The objective of our work is to study the application of clustering techniques for identifying homogenous groups of stakeholders that can be used as input to existing requirements selection and prioritization techniques.

Our stakeholders clustering technique takes as input stakeholders' values for a set of requirements to be evaluated, and generates as output a set of stakeholders groups together with the value of each requirement for the group. These groups and values would then be used by existing decision-making techniques to produce either a ranking of requirements based on their values, or a Pareto optimal front and fairness analysis diagram used for exploring tradeoffs. Our problem is very similar to the problem of market segmentation for product development and marketing [12]. In this area, one distinguishes between customer's characteristics that are product-independent such as his age, location and revenues, characteristics that are product-dependent such as his perceptions, benefits and loyalty for the product. Our stakeholders clustering approach consists of identifying stakeholders groups from their ratings which are product-dependent characteristics, instead of grouping them according to product-independent characteristics such as their job title or age.

The generated groups should be as homogenous as possible in the sense that all stakeholders within a group have roughly similar ratings for all requirements. When generating these groups, there is a conflict between minimizing the number of groups and maximizing their homogeneity. An extreme situation in which each stakeholder forms a single group would be very homogenous but would not help decision-making.

This paper describes our approach and illustrates its use on a pilot case study conducted at UCL where we explore the impact of using different group size and compare clustering approaches.

2. Using Cluster Analysis to Group Stakeholders: An Example

Similarity between stakeholders' ratings is determined by the distance between the stakeholders' ratings. A smaller distance implies a higher similarity between the ratings. If we have two ratings r_i and r_j from stakeholders i and j for the same requirement, the distance d between them is given as:

$$d=|r_i-r_j| \quad (1)$$

When we have n requirements R_1, R_2… R_n, this distance is computed as the Euclidean distance between the two sets of ratings for all n requirements:

$$d=\sqrt{[(r_{1i}-r_{1j})^2+(r_{2i}-r_{2j})^2+\ldots+(r_{ni}-r_{nj})^2]} \quad (2)$$

Our approach will use the weighted average linkage clustering algorithm [12]. This clustering algorithm is an agglomerative hierarchical one that helps us form a hierarchy of stakeholders with stakeholders with more similar ratings grouped lower in the hierarchy. The level of similarity decreases as we move up the hierarchy. Furthermore, the chosen clustering algorithm has other properties that are highly desirable in our context. It is a robust algorithm that reproduces the size and shape of the clusters as they occur in the dataset as closely as possible. It is not prone to reversal and chaining problems and it considers the size of the clusters when merging them [13]. The algorithm also has the benefit of being deterministic.

To test our approach, we have carried out a survey at UCL asking 50 potential stakeholders to rate 5 requirements R1, R2, R3, R4 and R5 for an online calendar on a 10 point scale. We obtained responses from 47 stakeholders, labeled S1 to S47. Our product-specific characteristics are the values of each requirement to the stakeholders. We have also gathered a few product-independent characteristics related to the stakeholder like position at UCL, number of years at UCL, and average number of hours spent online per day. Table 1 shows a sample of the data collected.

Table 1. Sample data collected from survey carried out at UCL.

Stakeholder	Position	Time spent on internet daily (hrs)	No. of years at UCL	Ratings				
				R1	R2	R3	R4	R5
S1	Technical/ Admin	6	4	5	3	10	10	10
S2	Research	10	7	3	10	3	3	3
..	………….	…..	….	….	….	….	….	….
S47	Academic	4	3	4	4	9	4	2

Figure 1 shows the dendogram representing the clusters generated by the weighted average linkage clustering algorithm on these ratings. A dendogram is a two-dimensional diagram that depicts how the agglomeration or division are done at the different stages of the cluster analysis [13]. The Y-axis depicts the distance among the ratings while the X-axis lists the stakeholders. As the topmost level of the hierarchy at cut off 10, we have a single cluster with all 47 stakeholders. When we move to the next level at a cut off of 9 we have 3 clusters. The first one consists of stakeholders S26 to S31 in the dendogram, the second one consists of stakeholders S2 to S34 and the third one consists only of stakeholder S9. As we decrease the distance along the Y-axis i.e. increase similarity, we have an increasing number of clusters which are smaller in size. We can see that in some cases, individual stakeholders are added to a cluster very late (at a greater distance) and these are outliers whose ratings could either be discarded or investigated in more depth depending on the importance and expertise of the particular stakeholder. One such example is stakeholder S9.

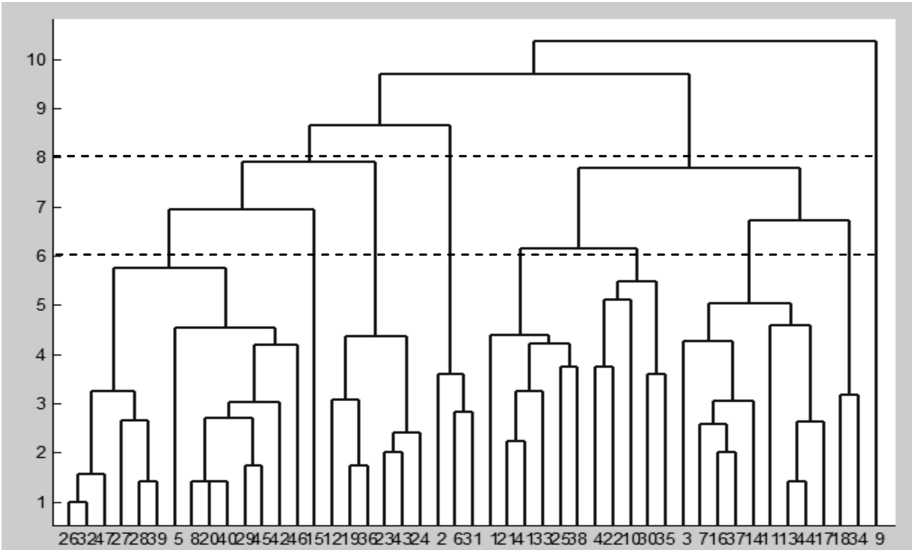

Fig 1. Dendogram for cluster analysis using weighted average linkage

When using our technique, requirements engineers need to define where they want to cut off the hierarchy to get the optimal clusters in their context. To explore the impact of such decisions, we have decided to cut off the dendograms at two places, that is, at distance 8 and 6. At cut off 8 we have 4 clusters and at cut off 6 we have 9 clusters. In each group, the value given to a requirement is taken to be the median value for all stakeholders in the group.

To assess the quality of the generated groupings and their associated requirements value, we compare for each stakeholder the distance between the requirement values for the group to which the stakeholders belong and the actual values given by the stakeholder. For comparison purposes we will also compute the distance of the actual ratings from the median of ratings with no clustering (i.e. all stakeholders are put in a single group) and when the stakeholders are grouped by their position at UCL. The results of this comparison are shown in Figure 2 for stakeholders S1 to S20 (the others are not shown to save space).

The figure shows that the median values used to represent the stakeholders with no clustering and grouping by role are generally farther from the actual ratings of the stakeholders than when clustering is used at all. With clustering, it can be seen that with a cut off at a lower difference results in a median value closer to the actual rating of the stakeholder. Stakeholder S9 is an outlier as it is a cluster on its own and the median values with clustering will be the actual ratings of the stakeholder.

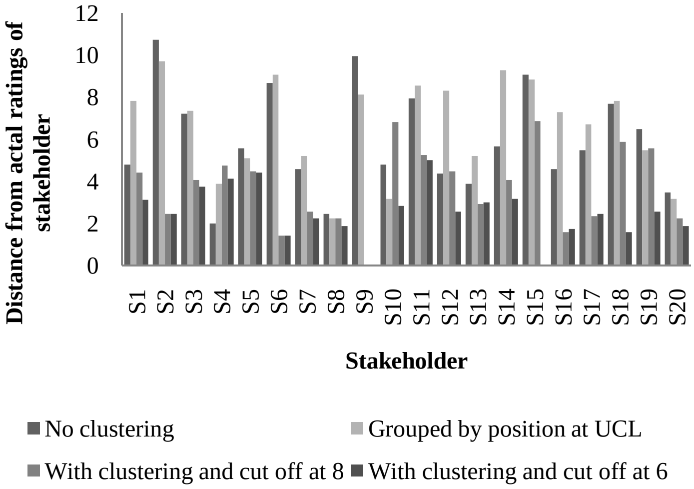

Fig 2. Distance between the actual ratings of stakeholders and the median ratings for their group using different cluster sizes and approaches.

To further demonstrate how the overall decision process is improved, we fed the median for the clusters (labeled G1 to G3) at cut off 8 into the Volere prioritization template [6] giving equal weight to all of the clusters. We have dropped the ratings of outlier S9, so we are left with 3 groups each having a weight of 33%. How to determine the weight to be given to each cluster is currently left to the decision maker. We intend developing techniques to help determining this weight based for example of the number of stakeholders it contains and on the composition of the group. Figure 3 shows the ranks of our requirements after running the Volere Prioritization using the median of ratings with and without clustering. The value and ranking of the requirements are different in the two approaches. For example, when stakeholders are first grouped in clusters, requirement R2 moves above R1 and R5 in the ranking order. As we have shown in Figure 2, this ranking is obtained by using median requirements values for each cluster that are closer to the actual values given by each stakeholder than the overall median value when no clustering technique is used. The identification of the clusters also allows us to explore different tradeoffs and analyze the fairness of the decisions.

Without Clustering

	Requirements Value	Rank
R1.	6	3
R2.	5	5
R3.	9	1
R4.	9	1
R5.	6	3

With Clustering at cut off 8

	G1	G2	G3	Weighted Sum Value	Rank
	0.33	0.33	0.33		
R1.	7	3	5	4.95	4
R2.	4	9	7	6.6	3
R3.	9	4	8.5	7.095	1
R4.	7	5	9.5	7.095	1
R5.	3	3	9	4.95	4

Fig 3. Extract from of results Volere Template with and without clustering

3. Conclusion

We have introduced the concept of segmentation of stakeholders using clustering analysis. The proposed technique was tested on a pilot case study at UCL and the results show that there is an improvement in overall closeness of the ratings used to make decisions when using cluster analysis. Our future work includes implementing a tool to enable requirements engineers to use this technique. We aim to enhance the technique with methods to describe the profiles of stakeholders belonging to a group and help decision makers assessing the weight to be assigned to each group. Divergences among stakeholders rating within a group might also be used to detect ambiguous requirements. Our immediate future plan is to test our technique on large independent data sets [3].

References

1. P. Laurent and J. Cleland-Huang, "Lessons Learned from Open Source Projects for Facilitating Online Requirements Processes," *Requirements Engineering: Foundation for Software Quality*, 2009, pp. 240-255.
2. S.L. Lim, D. Quercia, and A. Finkelstein, "StakeNet: using social networks to analyse the stakeholders of large-scale software projects," *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering - Volume 1*, Cape Town, South Africa: ACM, 2010, pp. 295-304.
3. S.L. Lim, "Social Networks and Collaborative Filtering for Large-Scale Requirements Elicitation." PhD Thesis. School of Computer Science and Engineering, University of New South Wales, 2010.
4. A. van Lamsweerde, "Reasoning About Alternative Requirements Options," *Conceptual Modeling: Foundations and Applications*, Essays in honor of John Mylopoulos, A Borgida, V. Chaudri, P. Giorgini, E. Yu (eds), Springer-Verlag LNCS, 2009, pp. 380-397 .
5. J. Karlsson and K. Ryan, "A cost-value approach for prioritizing requirements," *IEEE software*, vol. 14, 1997, pp. 67–74.
6. S. Robertson and J.C. Robertson, *Mastering the Requirements Process*, Addison Wesley, 2006.
7. Karl Wiegers E. First things first: Prioritizing requirements. *Software Development* 1999; 7(10):24–30.
8. IBM - Project, product and portfolio management software - Rational Focal Point - Software. http://www-01.ibm.com/software/rational/
9. Y. Zhang, M. Harman, and S.A. Mansouri, "The multi-objective next release problem," *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, London, England: ACM, 2007, pp. 1129-1137.
10. A. Finkelstein, M. Harman, S.A. Mansouri, J. Ren, and Y. Zhang, ""Fairness Analysis in Requirements Assignments," *Proceedings of the 2008 16th IEEE International Requirements Engineering Conference*, IEEE Computer Society, 2008, pp. 115-124.
11. Y. Zhang, "Multi-Objective Search-based Requirements Selection and Optimisation," PhD Thesis. Department of Computer Science, King's College London, 2010.
12. M. Wedel and W.A. Kamakura, *Market segmentation: conceptual and methodological foundations*, Springer, 2000.
13. B. S. Everitt, S. Landau and M. Leese, *Cluster Analysis*, Hodder Arnold, 2001.