Embracing Composite Metrics in Software Experiments

Mohamed El-Attar
Department of Information and Computer Science, King Fahd University of Petroleum and Minerals, Dhahran 31261, Kingdom of Saudi Arabia
melattar@kfupm.edu.sa

Ravi Inder Singh[1], James Miller[2]
Department of Electrical and Computer Engineering, Edmonton, Alberta T6G 2VB, Canada
[1]homme.sauvage@gmail.com, [2]jimm@ualberta.ca

***Abstract*—Traditionally most Software Engineering experiments tend to formulate hypotheses and analyze an independent variable or a series of independent variables. This approach greatly reduces the type of research questions which can be explored. In addition, most Software Engineering situations are highly complex with many intertwined or ill-defined concepts, processes and "objects". Hence, the question arises: are independent variables really sufficient for describing Software Engineering situations? This paper argues that the community needs to consider fuzzier models for Software Engineering artifacts, especially it recommends using composite indices as a mechanism to allow the greater exploration of the experimental design space. However, this extension is not without its risks; and hence in conjunction, it explains how to utilize analysis safeguards (sensitivity and uncertainty analysis) to explore any effects introduced when utilizing experimental formulations with composite metrics.**

***Index Terms*— Composite Metrics, Sensitivity, Uncertainty, Case Study**

I. INTRODUCTION

[1]Software engineering experiments have a number of issues most of which are based around the lack of absolute definitions of many of the variables either "controlled" or "analyzed" in its experiments. Unlike variables and definitions in the hard sciences, concepts in Software Engineering are defined by man and tend to be virtual in nature. This lack of formal and universal description results in quantities that are dimensionless; and hence, we often have no real basis for understanding which concepts are truly the same "type" (in a measurement sense) and which are only related.

Consider any experiment or case study which investigates the number of defects present in a document. These experiments typically count the number of defects either found by various techniques or tools. The implicit assumptions behind these "counts" are that all of these defects are of the same type (to allow additive summation) and that they all have the same value (commonly a unit value – interval scale) and that the count can be meaningfully interpreted when it is zero (rational zero – ratio scale). This allows researchers to undertake a large array of differing analysis techniques – however, are the assumptions really justified?

Consider an industrial debugging process; typically, upon finding a bug, a practitioner describes it to allow explicit decisions upon the "fate" of the bug. As part of the description, the practitioner often includes their assessment of the severity of the bug. For example, D0-248B[2] (Software Considerations in Airborne Systems and Equipment Certification) characterizes defects as:

1. **Catastrophic:** Defects that could (or did) cause disastrous consequences for the system.
2. **Severe:** Defects that could (or did) cause very serious consequences for the system.
3. **Major:** Defects that could (or did) cause significant consequences for the system.
4. **Minor:** Defects that could (or did) cause small or negligible consequences for the system.
5. **No Effect:** Defects that can cause no negative consequences for the system.

Clearly, from these descriptions it is difficult to believe that DO-248B considers that these 5 types of defects possess the same value. Hence, should researchers consider them to be of different types?

Schemas like Orthogonal Defect Classification (ODC) [3] and Defect Origins, Types, and Modes [12] classify defects into alternative dimensions or types. In other situations, defects are classified by their cause (Root cause analysis) or by the phase in the life-cycle where they are injected. While, it is unlikely that we will ever have a single definition of the numerical properties of a defect, a case can be easily made that treating them as a single homogenous numerical-definition is not the only option.

Other common ideas in IT are by definition an amalgamation of concepts which are clearly distinct. Consider, ISO 9126 as a potential definition of software quality. The standard states that quality is composed of: functionality, reliability, usability, efficiency,

Corresponding author: Mohamed El-attar (melattar@kfupm.edu.sa)

[2] DO-248B is the final report for clarification of DO-178B

doi:10.4304/jsw.7.7.1664-1676

maintainability, portability. Therefore, if we want to talk about total quality we need to be able to construct something like:

Total Quality = Quality (functionality) + Quality (reliability) + Quality (usability) + Quality (efficiency) + Quality (maintainability) + Quality (Portability)

So how can we construct such a variable as Total Quality? How can we safely aggregate the sub-quality expressions? How can we understand the relative impact of the sub-quality variables upon the total quality variable? Further, the characteristics themselves are further composed of sub-characteristics; e.g. functionality is the composition of suitability, accuracy, interoperability, compliance, security. Once again, the average practitioner may well ask – is security composed of anything? [1]

Software Engineering differs from many other scientific disciplines due to its limited number of physical components and concepts. Physical components allow us to invoke ideas from the "hard sciences" to produce precise definitions and relationships. This provides a solid basis for many empirical investigations and for them to have unique interpretations. In contrast, Software Engineering often deals with attributes which are more meta-physical than physical, have imprecise definitions which vary over time, domain and problem statement; and often have no "mechanism" to allow us to either observe or measure them directly.

Consider the "definition" of Trust taken from [2]. This definition represents a meta-analysis of the literature on the definition of Trust. Points to note are:

- The definition considers the attribute only within a limited domain (e-commerce). So what do the results imply for the definition in say organisational settings?
- To produce the definition, the researchers considered any on-line activity as a proxy for e-commerce behaviour. Using proxies in Software Engineering is extremely common because of imprecision of definitions – however, it is not without substantial risk.
- Trust seems to be defined in terms of a large number of attributes. Anything in Figure 1 with a direct link to trust has potentially a direct relationship. The number on the link indicates the number of studies which "found" this relationship. The number in the node is the number of studies which actively considered this attribute when exploring this question. Some researchers even defined attributes as themselves! See the weights on the self-references from a node to itself.
- These defining attributes are highly inter-related with other attributes which define Trust. Forget linearity.

These defining attributes also have complex, imprecise definitions based upon a network of attributes. Each of these attributes within this network of attributes has in turn a definition based upon a network of attributes, and so on. As can be seen in Figure 1, most of the variables surrounding trust are ill-defined. Clearly traditional statistical experiments and analysis are not going to work here; we require a more open-ended framework with a much greater emphasis on exploration to start unravelling this picture.

Clearly, Software Engineering is a topic dominated by terms which are compositions of concepts which are themselves a composition of concepts, which are …. So if Software Engineering is dominated in such a fashion – why are Software Engineering experiments not dominated by composite variables and appropriate analysis of such variables? This paper seeks to address this topic. By extending the analysis of a previous paper by the authors [9], we seek to introduce to the field a simple basis for constructing such variables; and an analysis approach to allow them to be utilised relatively safely.

The remainder of the paper is structured as follows: Section 2 seeks to explain situations where compound variables or indices are not required – moving to compound or composite variables should only be undertaken when appropriate. Section 3 provides a description of the conditions under which it is reasonable to use compound or composite variables. Sections 4 and 5 provide a practical illustration of using such variables. Specifically, Section 4 presents an overview of the results from [9] which is our starting point for introducing compound variables. Section 5 provides a case study in using compound indices by expanding the results in Section 4 to provide a richer set of analysis and the exploration of additional hypotheses. Section 6 concludes the paper and seeks to provide guidelines on using compound indices.

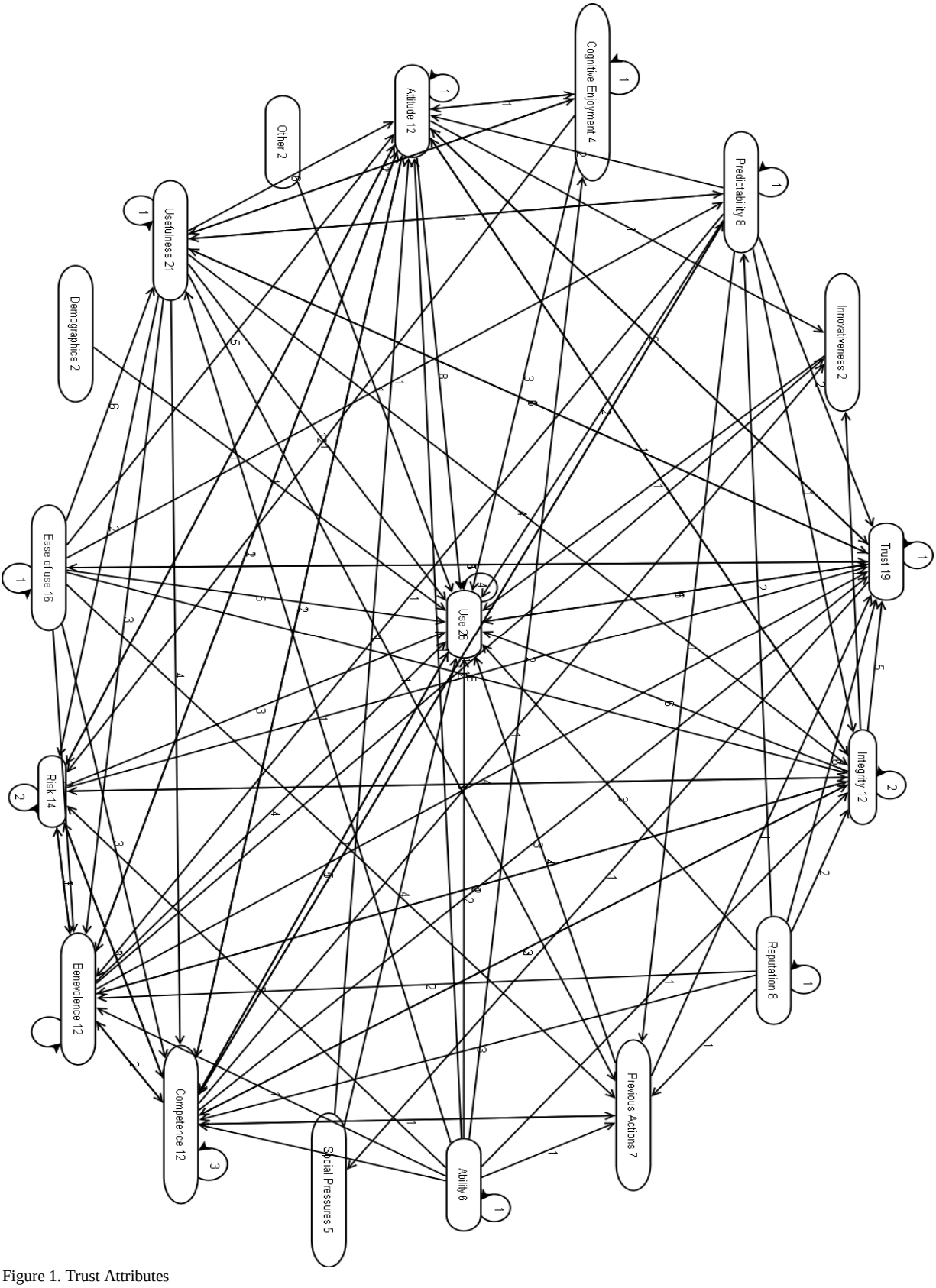

Figure 1. Trust Attributes

II. Exploration Analysis and Composite Variables

As was seen in Section 1, many "interesting" empirical questions do not fall within the traditional definition or the domain of Neyman-Pearson type significance testing [15]. Empirical thinking should not simply involve the measurement of the error and inserting this figure into a test. Good empirical investigation seeks to explore the error from many angles. In an ideal situation, we may have been successful in conducting an experiment with an extreme low error. Via careful selection of experimental parameters (choice of task, choice of subjects, motivation of subjects, providing clear guidance, etc), we can seek to minimize this error. However, in many situations, we will have limited insight into how successful we have been; and hence, it is essential to explore the data from a variety of angles looking for explanations of what really happened. Inductive inference has no single best solution – many alternatives exist; empirical analysis involves analyzing the problem at hand and selecting the "best" set of tools based upon that analysis.

Consider the following scenario adapted from Gigerenzer et al. [10]. A typical application of Neyman-Pearson type significance testing is in quality control in a manufacturing setting. Imagine we are producing widgets with a mean diameter of 5 mm (H1) as optimal and 7 mm (H2) as dangerous and hence unacceptable. From experience, it is known that the random fluctuations of diameters are approximately normally distributed and that the standard deviations do not depend on the mean. This allows the experimenter to determine the sampling distributions of the mean for both hypotheses. They consider accepting H1 while H2 is true (Type II error) to be the most serious error because it may cause harm to the users and to the company's reputation. So they set β = 0.1% and α = 10%. They now calculate the required sample size n of widgets that must be sampled every day to test the quality of the production. When they accept H2, they act as if there was a malfunction and stop production, but this does not imply that they believe that H2 is true. They know that they must expect a false alarm, on average, in 1 out of 10 days in which there is no malfunction.

Does this quality experiment sound like the experiments outlined in Section 1. It is argued – no! Here the definition of "total quality" is very simple; it needs to consider only one variable which can be measured directly. The measure has a precise definition given by the units of measurement, which are absolute. The experiment has an instrument which can measure the variable directly and this instrument, via calibration, etc, has known characteristics and is guaranteed (when operating correctly) to have a very high level of accuracy. The experiment is able to set two completing hypotheses; one which defines acceptable behavior, and one that defines unacceptable behavior. These in conjunction with definitions of α and β define a rejection region for each hypothesis. Further, the experimenter has the luxury of an also infinite sample size, the factory produces large volumes of widgets, allowing α, β and n to be set in line with the economic situation and in a fashion where the risks of encountering an error in inference are balanced against this economic situation.

It is argued that in such well–defined situations that traditional Neyman-Pearson type significance testing provides an excellent tool-set to analyze the behavior of the situation. However, as we move away from this type of situation; and encounter ill-defined variables (variables which are potentially dependent, variables which can't be measured directly, etc.), we need to augment this tool set with more exploratory analyze where we aggregate variables and look at the implications.

III. An Overview Of Composite Variables

Any useful composite variable has to be defined using a sound methodology. Typically, there are several stages for the construction of composite variables:

- **Deciding on the phenomenon to be measured.** Would it benefit from the use of composite variables? For example, it is argued that the above manufacturing scenario would not.
- **Selection of sub-variables.** A sound casual argument or empirical evidence is required as to identify which sub-variables are relevant to the meta-variable of interest. In general, there is no completely objective way of selecting these sub-variables.
- **Assessing the quality of the data.** There needs to be high quality data for all the sub-variables; otherwise, the analysis will be meaningless. If all of the sub-variables are believed to contain large errors and then meta-variables will have extremely large errors rendering the analysis useless. Careful use of uncertainty and sensitivity analysis protects against this issue.
- **Assessing the relationships between sub-variables.** Analyzing the sub-variables for correlations is important in many situations. Since, the sub-variables will be aggregated in some fashion, independence is important to avoid "double-counting".
- **Normalizing and weighting of the variables.** Many methods for normalizing and weighting the sub-variables exist. The selection of the appropriate methods depends on the situation, the collection process and ultimately the data.
- **Uncertainty and Sensitivity Analysis.** Changes in the weighting system and the choice of mechanisms to aggregate the sub-variables will affect the results obtained from the analysis. It is important to test the degree of sensitivity to fluctuations in the sub-variables; and avoid reporting results which are highly sensitive to small changes in the construction of the composite variable. The value of the composite variable should always be analyzed to provide some form of confidence bound (e.g. under what ranges of values does this result remain constant?) upon the result.

Clearly, one could write a textbook on these topics; instead, we have chosen to illustrate the use of composite variables in Software Engineering via a single case study. Specifically, the illustration will utilize a formal subject-based experiment. Further, it will demonstrate how to extend Neyman-Pearson type significance testing to accommodate composite variables and composite questions. No claim is made that the original experiment is a perfect example of a situation where using composite variables has no downside. In fact, the limited sample size in the original experiment must be considered an issue. However, it will be demonstrated that via the construction of composite variables, we are able to explore a wider range of questions; while introducing sufficient safeguards to (hopefully) stop us from over-interpreting the results.

IV. CASE STUDY

In [9] the authors undertook a pilot experiment and a main experiment to assess the impact of utilising a structuring technique upon the quality of Use Case (UC) models. The reader is referred to this paper for details. The paper views quality in this context as having several distinct concepts; see Table 1 for details.

TABLE 1. QUALITY ATTRIBUTES OF A USE CASE MODEL

Quality Attribute	Definition
Consistency	The UC diagram must conform to the concepts contained in the UC descriptions and vice versa. Consistent facts and information must be present across UC descriptions. If a UC model contains more than one UC diagram, consistency must exist between UC diagrams with respect to elements that they depict.
Completeness	The underlying requirements must correctly be represented by the UC diagram and textual descriptions. This means that all information and facts that are expected to be in the UC descriptions and diagram must be present.
Fault-Free	The UC diagram and descriptions must not contain any information or facts that are incorrect, which misrepresent the underlying requirements.
Analytical	The model should be analytical, meaning that it should only describe what the system should do. This includes the exclusion of any design or implementation decisions, including interface details. Except those explicitly defined by the customer.
Understand-ability	The model must be presented in a readable form. The information contained in the UC descriptions must be precise and unambiguous. The model should also not contain repeated information as this may lead to confusion. All stakeholders must share a common understanding of the presented functional requirements.

In line with common practice in subject-based Software Engineering experiments, the concepts in [9] are treated separately and a unique hypothesis is constructed for each distinct type of quality. Hence, hypotheses with regard to the overall impact on total quality, etc are not explored, as this requires an integration of the quality types into a single meta-concept. The authors believe that this type of omission is really a failure to explore the analysis space and that the topic must embrace compound indices to allow it to explore a richer set of questions. However, this exploration is not without issues; and Section 5 outlines an approach for handling compound indices in subject-based experimental situations when using inferential statistics.

A. Overview of Current Analysis

The only independent variable of the experiment was the use of the structuring language named SSUCD (Simple Structured Use Case Descriptions) [8]. The use of SSUCD was hence compared to the use of Unstructured Natural Language (UNL). The experiment involved a total of 34 graduate Software/Computer/Electrical Engineering students, who were divided into two groups of 17 subjects each. The subjects were required to construct the use case model of an Airline Ticketing system [17] and a Banking system [11] using SSUCD and UNL. The experiment deployed a 2 x 2 partial factorial design with repeated measures to mitigate the effect of individual and group abilities. The schedule of the experimental tasks is outlined in Table 2.

TABLE 2. SCHEDULE OF EXPERIMENTAL TASKS

	Group A		Group B	
Week 1	Introduction to UC modeling - 2 lectures (approx. 2 hours total)			
Week 2	UC modeling practice using UNL and SSUCD – 3 lectures (approx. 3 hours)			
Week 3	UNL	Develop Airline Ticketing system	SSUCD	Develop Airline Ticketing system
Week 4	SSUCD	Develop Banking system	UNL	Develop Banking system

The subjects' use case models were evaluated with respected to each of the quality attributes shown in Table 1. The raw scores per quality attribute are shown in Tables 31→35 (see Appendix A). The Mann-Whitney U statistic (of the 1st sample) was used to test for the differences between the [21]. The Hodges-Lehman method was used to compute the confidence intervals around the medians at the standard 95% level. For major results from statistical significance testing, an estimate of the difference between the two groups was provided by estimating the associated effect size using Cliff's delta [4-6]. For two samples, if the confidence interval of Cliff`s Delta includes zero, then the populations are considered equal – that is, insufficient difference exists to distinguish between the samples. If the confidence interval excludes zero then sufficient information exists to distinguish the samples. In this experiment, if the confidence interval included only positive numbers then SSUCD > UNL (favoring SSUCD). If it only contained negative numbers

then UNL > SSUCD (favoring UNL). The analysis performed investigates the effects of the treatment variables, experimental artifacts and groups' abilities in isolation, with respect to each quality attribute. Table 3 presents the results from investigating the effect of using SSUCD vs. UNL on each system with respect to the quality attributes (statistically insignificant results were omitted from the Table 3.) Table 4 presents the results obtained for the Airline Ticketing system vs. Banking system with respect to each quality attributes individually. Again, only statistically significant results are shown. Note that there were no statistically significant results obtained from investigating the performance of Group A vs. Group B with respect to all quality attributes. The statistically significant results presented in Tables 3 and 4 are further confirmed in favor of SSUCD as the confidence interval around Cliff's Delta includes only positive values as shown in Tables 5 and 6, respectively.

TABLE 3. MANN-WHITNEY TESTS FOR THE SSUCD VS. UNL RESULTS

System	Technique	Rank sum	Mean rank	U	Difference between medians	95.2% CI	Mann-Whitney U statistic	1-tailed p
Inconsistencies								
Airline Ticketing	**SSUCD** **UNL**	362.5 232.5	21.32 13.68	79.5 209.5	1.0	0.0 to +∞	79.5	0.010
Banking	**SSUCD** **UNL**	337.5 223.5	21.09 13.15	70.5 201.5	1.0	0.0 to +∞	70.5	0.010
Completeness								
Airline Ticketing	**SSUCD** **UNL**	364.0 231.0	21.41 13.59	78.0 211.0	1.0	0.0 to 2.0	78	0.018
Understandability								
Airline Ticketing	**SSUCD**	385.5	22.68	56.5	2.0	1.0 to 3.0	56.5	<0.01

TABLE 4. MANN-WHITNEY TESTS FOR THE AIRLINE TICKETING SYSTEM VS. BANKING SYSTEM RESULTS

System	Rank sum	Mean rank	U	Difference between medians	95.2% CI	Mann-Whitney U statistic	2-tailed p
Fault Free							
Airline Ticketing	1368.0	40.24	349.0	0.125	0.0 to 0.250	349	0.007
Banking	910.0	27.58	773.0				
Understandability							
Airline Ticketing	1375.0	40.44	342.0	0.250	0.083 to 0.333	342	0.006
Banking	903.0	27.36	780.0				

TABLE 5.
CLIFF'S DELTA FOR THE SSUCD VS. UNL RESULTS

System	Cliff's delta ($\hat{\delta}$)	Variance	Confidence Interval around delta ($\hat{\delta}$)	
			Max.	Min.
Inconsistencies				
Airline Ticketing	0.450	0.030	0.673	0.112
Banking	0.764	0.028	0.783	0.435
Completeness				
Airline Ticketing	0.460	0.030	0.680	0.122
Understandability				
Airline Ticketing	0.609	0.024	0.737	0.324

TABLE 6.
CLIFF'S DELTA FOR THE SSUCD VS. UNL RESULTS

Quality Attribute	Cliff's delta ($\hat{\delta}$)	Variance	Confidence Interval around delta ($\hat{\delta}$)	
			Max.	Min.
Fault-Free	-0.378	0.019	-0.090	-0.608
Understandability	-0.390	0.021	-0.086	-0.627

II. EXTENDING THE RESULTS USING COMPOSITE INDICES

In this section, we will explore the results further to provide some illustrative experiment-wide numerical statements about the study. Our exploration is based upon the construction of a number of composite indices, by combining our sub-indicators (the individual quality characteristic performances) into a single index on the basis of representing an implicit underlying model in this paper (that of the total performance of any individual subject in either of the tasks). Clearly, this approach requires the reader to carefully consider its results as it effectively summarizes complex, multi-dimensional issues into simple numerical statements and hence it is easy to over-interpret the output resulting from subsequent analysis of these numerical statements.

In common with current practice, we will construct composite indices by a weighted combination of normalized sub-indicator values [14, 18]; specifically, we will construct composite indices of the form:

Let SI represent any arbitrary sub-indicator:

$$Score_i = w_1 * F(SI_{1,i}) \nabla w_2 * F(SI_{2,i}) \nabla w_3 * F(SI_{3,i}) \nabla .....$$

Where w_i is a weighting factor; *F()* is a normalization function; ∇ is an aggregation operator; and w_1 has a higher precedence than w_2. Unfortunately, there is no universal formulation for these three terms; and hence the following discussion is our rational for our selected instantiations.

Normalization Function: A normalization function is required to stop us combining "apples" and "oranges"; and provides effective protection against data misuse, such as Simpson's paradox [23]. It is important to select the appropriate normalization procedure with reference to both the data properties and the theoretical framework. This action requires scrutiny of the data set for sub-indicators. Even after scrutiny, the selection of a procedure is not obvious; there are still risks to be worked through. Recognizing and working through these risks is a fundamental exercise in this article. In due time a normalization procedure must be selected. We selected re-scaling. There are five other normailization procedures that could have been used; each has its own variations in application and of course, its own risks. We selected re-scaling as a normalization procedure and were cognizant of its risks; it was the same procedure used in the case study. The re-scaling normalization procedure is often encountered in the literature and we have followed the same procedure. Note that outlining the complete set of risks for each normalization procedure is lengthy and beyond the scope of this paper.

As previously mentioned, for the type of data and analysis within this study, two options basically exist: standardization or re-scaling. Standardization approaches require the estimation of several parametric descriptors for the sub-indicators. However, as indicated earlier, our sub-indicators have neither theoretical arguments nor numerical support for the statement that they are sampled from a parametric distribution; and hence, re-scaling becomes the only viable option. In fact, during much of our exploratory analysis, many common approaches and techniques are not available due to various distribution requirements that they possess. For the sake of brevity, we will in general not discuss when individual approaches are not suitable for this reason.

Re-scaling can be considered as the non-parametric analog to standardization. Re-scaling simply transforms all sub-indicators into an identical range (0,1). Specifically, using the normalization function:

$$F(j,i) = \frac{SI_{j,i} - \min(SI_j)}{range(SI_j)}$$

A disadvantage of this approach is that the minima and maxima might be unreliable outliers, and have a distortion effect on the normalized indicator. On the other hand, for sub-indicator values lying within a small interval, this method increases the effect of the indicator on the composite index. No obvious evidence exists in our data that these effects are extreme, but given their nature it is impossible to say that no impact exists; and they should be considered as a threat to the numerical validity of the results from this section. Finally, before the normalization function, we transform our data to have a common orientation. The transformed sub-indicators use a positive only scale, where a 'low' value implies a 'poor' performance and a 'high' value implies a 'good' performance.

Weighting factor: The weighting factor basically describes the relative importance of each of the sub-indicator terms. Normally this is unknown and these values are estimated by domain experts; e.g. Saisana *et al.* [19] outline an estimation approach, where expert opinion is collected and analyzed using budget allocation and analytical hierarchy process approaches. However, we believe that these types of approaches are unlikely to be fruitful within our domain, and hence, we take a simpler approach. Initially, all of the weighting terms have been set to have a unit value; subsequently all of the weights are varied to form bounds upon the results; this process forms the core of our approach to uncertainty analysis which is discussed in the next section.

Aggregation Operation: Here we select a suitable basis for combining the weighted normalized sub-indicators. Again, we believe that the problem leads to a unique choice of operator, addition. (This decision is revisited in Section 5.2) Hence we form a simple linear additive statement. Other alternatives include geometric aggregation or non-compensatory multi-criteria analysis [13], however neither of these alternatives provide a suitable formulation for subsequent analysis in terms of statistical significance testing and effect size estimation.

This formulation implies that ideally the sub-indicators should be independent. Clearly, this technical requirement is impossible to meet in this "style" of problem, as the sub-indicators are just different aspects of the performance of an individual subject within the study. Hence, dependence between the sub-indicators is an inevitable feature in any formulation of this concept. However, given this limitation no clear course of action exists. One may view the dependence among the sub-indicators as something to correct for; For example, by making the weight for a given sub-indicator inversely proportional to the arithmetic mean of the coefficients of determination for each correlation that includes the given sub-indicator. On the other hand, practitioners of multi-criteria decision analysis would tend to consider the existence of the dependences as a feature of the problem, not to be corrected for, as correlated indicators may indeed reflect non-compensable different aspects of the problem. We explore this issue principally, but implicitly, by our uncertainty analysis approach, which explores the impact of the weighting factors. However, in general, we resist the temptation to attempt to correct the dependences as we view them as non-compensable features of the domain. Keep in mind though, that this aspect undoubtedly should be viewed as a threat to the validity of the numerical results within this section.

A. Uncertainty and Sensitivity Analysis

When constructing our composite indexes we make three decisions; in the above section, we argue that two of the decisions (normalization and aggregation) have unique definitions within this context. Hence, we will only explore the impact of the remaining definition

(weighting). In addition, we examine the impact of the input sample, as it is drawn without reference to a sampling frame, by neither a randomized and representative sampling procedure, and without specific targets at controlling Type II errors, and hence, it must be considered as less than ideal. We regard these explorations as approaches to uncertainty [7] and sensitivity analysis [20] respectively. We explore these two cross-validation approaches only within the concept of our data analysis goals. Within this experiment, we are principally interested in a binary decision (significant or non-significant); and hence we utilize this fact to shorten the analysis approach to only yield additional insight into these decisions. That is, we only really consider two questions:

- Can changes to the weighting factors change the result of the statistical significance tests, where change is defined as crossing the binary decision threshold when compared with the results from our control result (with every weight factor considered equal)?

- Do particular input data items overly influence the result of the statistical significance test? That is, if they are withdrawn from the data set, does the binary decision change?

More specifically our two approaches to cross-validation are as follows:

Sensitivity Analysis: Again, we can avoid more generic re-sampling statistical approaches [22]; and replace it with an exhaustive search. Here, we withdraw every permutation of input pairs and see if the binary decision changes. (In reality, only the most extreme pair needs to be investigated.) If the decision does not change, we withdraw every permutation of two input pairs, etc. We stop when either the binary decision changes or the data set is exhausted.

Sensitivity analysis will be presented in tables that contain four main columns. The first column will contain the original *P* value obtained from the Mann-Whitney statistical test performed on the original values. The second column shows the change in statistical significance after *n* pairs are withdrawn. The third column indicates the number of pairs that were withdrawn in order to cause a change in the statistical significance (from significant to insignificant or vice versa). The fourth column indicates the category in which pairs were withdrawn in favor of.

Uncertainty Analysis: Given our limited requirements, we can avoid using Monte Carlo sampling of the factor (and Monte Carlo Permutation tests [16]) and simply replace them with an exhaustive search. For each weighting factor, we independently vary it by an order of magnitude in each direction (i.e. [0.1, 10]) and record when, and if, the binary decision changes.

Uncertainty analysis will be presented in tables that contain two main columns. The first column shows the range that a weighting, for a certain quality attribute, can change while maintaining the statistical significance (or insignificance) indicated in the second column observed in the original analysis.

SSUCD vs.UNL – All attributes combined

Table 7 shows the overall quality achieved by the subjects when using SSUCD and UNL with respect to both systems. As shown in Table 8, statistical significance was observed with respect to the overall quality achieved by the subjects with the Airline Ticketing system only. Sensitivity analysis of the results (Table 9) and Cliff's Delta (Table 10) both indicate that subjects performed better with SSUCD over their UNL counterparts with respect to the Airline Ticketing System. The uncertainty analysis (Table 11) performed indicates that there was no single quality attribute contributing significantly the most towards the statistical significance observed with respect to the Airline Ticketing system.

TABLE 7. DESCRIPTIVE STATISTICS OF THE RESULTS FOR BOTH SYSTEMS

System	Technique	n	Median	IQR	95% CI of Median
Airline Ticketing	SSUCD	17	4.342	0.474	4.076 to 4.550
	UNL	17	3.922	1.180	2.963 to 4.143
Banking	SSUCD	16	3.697	0.788	3.250 to 4.196
	UNL	17	3.446	0.524	3.113 to 3.637

TABLE 9. SENSITIVITY ANALYSIS OF THE RESULTS FOR BOTH SYSTEMS

Overall Quality	Original P value before pairs removal	Change in P value after pairs removal	# of pairs removed	In favor of
Airline Ticketing	0.005	Insignificant	3	UNL
Banking	0.130	Significant	8	UNL
		Significant	1	SSUCD

TABLE 8. MANN-WHITNEY TEST FOR BOTH SYSTEMS

System	Technique	Rank sum	Mean rank	U	Difference between medians	95.2% CI	Mann-Whitney U statistic	2-tailed p
Airline Ticketing	SSUCD	380.0	22.35	62.0	0.589	0.171 to 1.178	62	0.005
	UNL	215.0	12.65	227.0				
Banking	SSUCD	314.0	19.63	94.0	0.320	-0.089 to 0.726	94	0.130
	UNL	247.0	14.53	178.0				

TABLE 10. CLIFF'S DELTA FOR BOTH SYSTEMS

System	Cliff's delta ($\hat{\delta}$)	Variance	Confidence Interval around delta ($\hat{\delta}$)	
			Max.	Min.
Airline Ticketing	0.571	0.026	0.802	0.191
Banking	0.313	0.039	0.628	-0.092

TABLE 11. UNCERTAINTY ANALYSIS OF THE RESULTS FOR BOTH SYSTEMS

System	Quality Attribute	Assigned Weight	P value
Airline Ticketing	*Inconsistencies*	0.1 ↔ 9	Significant
		10	Insignificant
	Completeness	0.1 ↔ 10	Significant
	Fault-Free	0.1 ↔ 4	Significant
		5 ↔ 10	Insignificant
	Analytical	0.1 ↔ 8	Significant
		9 ↔ 10	Insignificant
	Understandability	0.1 ↔ 10	Significant
Banking	*Inconsistencies*	0.1 ↔ 2	Insignificant
		3 ↔ 10	Significant
	Correctness	0.1 ↔ 10	Insignificant
	Incorrectness	0.1 ↔ 0.5	Significant
		1 ↔ 10	Insignificant
	Analytical	0.1 ↔ 10	Insignificant
	Understandability	0.1 ↔ 10	Insignificant

Airline Ticketing System vs. Banking System

Table 12 shows the overall quality achieved by the subjects with each system. As shown in Tables 13 and 14, no statistical significance was observed between the overall quality achieved with the Airline Ticketing system and the Banking system. Sensitivity analysis of the results show that statistical significance will be observed if only one pair was removed in favor the Airline Ticketing system in comparison to twelve pairs removed in favor of the Banking system (Table 15). Uncertainty analysis (Table 16) shows that there is no single quality attribute that can lead to statistical insignificance being observed between the two systems, which indicates that the subjects performed relatively close with respect to each quality attribute

TABLE 12. DESCRIPTIVE STATISTICS OF THE RESULTS

System	n	Median	IQR	95% CI of Median
Airline Ticketing	34	3.951	0.813	3.727 to 4.284
Banking	33	3.601	0.667	3.321 to 3.839

TABLE 13.
CLIFF'S DELTA – AIRLINE TICKETING VS. BANKING

Cliff's delta ($\hat{\delta}$)	Variance	Confidence Interval around delta ($\hat{\delta}$)	
		Max.	Min.
-0.270	0.023	0.039	-0.532

TABLE 15. SENSITIVITY ANALYSIS OF THE RESULTS

	Original P value before pairs removal	Change in P value after pairs removal	# of pairs removed	In favor of
Overall Quality	0.057	Significant	1	Airline Ticketing System
		Significant	12	Banking System

TABLE 16. UNCERTAINTY ANALYSIS OF THE RESULTS

Quality Attribute	Assigned Weight	P value
Inconsistencies	0.1 ↔ 10	Insignificant
Completeness	0.1 ↔ 10	Insignificant
Fault-Free	0.1 ↔ 10	Insignificant
Analytical	0.1 ↔ 10	Insignificant
Understandability	0.1 ↔ 10	Insignificant

Group A vs. Group B

Table 17 shows the overall quality achieved by each group. As shown in Tables 18 and 19, there is no statistical significance observed between the groups. Sensitivity analysis (Table 20) of the results indicate that statistical significance will be observed after the scores of 5 subject pairs have been removed in favor of Group A or Group B. Uncertainty analysis performed (Table 21) indicate that each group performed at a close level with respect to each quality attribute. This further confirms the analysis performed previously in Section 4.1.3, which indicates that both groups have proximate capabilities.

TABLE 17. DESCRIPTIVE STATISTICS OF THE RESULTS

Group	n	Median	IQR	95% CI of Median
SSUCD	34	3.570	0.854	3.327 to 3.918
UNL	33	3.644	1.254	3.250 to 4.000

TABLE 19. CLIFF'S DELTA

Cliff's delta ($\hat{\delta}$)	Variance	Confidence Interval around delta ($\hat{\delta}$)	
		Max.	Min.
-0.031	0.024	0.263	-0.320

TABLE 20. SENSITIVITY ANALYSIS

	Original P value before pairs removal	Change in P value after pairs removal	# of pairs removed	In favor of
Overall *Quality*	0.003	Significant	5	Group A
		Significant	5	Group B

TABLE 21. UNCERTAINTY ANALYSIS OF THE RESULTS

Quality Attribute	Assigned Weight	P value
Inconsistencies	0.1 ↔ 10	Insignificant
Correctness	0.1 ↔ 10	Insignificant
Incorrectness	0.1 ↔ 10	Insignificant
Analytical	0.1 ↔ 10	Insignificant
Understandability	0.1 ↔ 10	Insignificant

TABLE 14. MANN-WHITNEY TEST

System	Rank sum	Mean rank	U	Difference between medians	95.2% CI	Mann-Whitney U statistic	2-tailed p
Airline Ticketing	1307.5	38.46	409.5	0.333	-0.006 to 0.607	409.5	0.057
Banking	970.5	29.41	712.5				

TABLE 18. MANN-WHITNEY TEST FOR SSUCD VS. UNL

Group	Rank sum	Mean rank	U	Difference between medians	95.2% CI	Mann-Whitney U statistic	2-tailed p
SSUCD	1176.0	34.59	541.0	0.050	-0.307 to 0.439	541	0.802
UNL	1102.0	33.39	581.0				

B. Meta-level Analysis

Our experiment can be viewed as a pair of experiments. In this section, we explore the aggregation of the results from the 'two' experiments. Clearly, this has parallels with meta-analytic procedures (fixed-effects models) [22]; however, here we have the raw data available not just the summary statistics. As within the previous section, we believe that this is an imprecise process and hence the issue is best framed in an exploratory nature. We view this process, and the results, as containing a significant level of risk as clearly the experiments are far from independent and hence have a significant number of sources of potential common bias. Hence, we must urge caution when interpreting the results from within this section.

Aggregation for Each Quality Characteristic

Here we merge (as aggregation) the results from the individual tasks into a single meta- statement. The results are shown in Table 22. To allow for significance testing, the merged scores need to be ranked; and hence to avoid comparing incompatible types, we again normalize, or more precisely rescale, every score before comparison. Within this framework, weightings have no real meaning, and hence uncertainty analysis is not applied; however, sensitivity analysis exists as before. Again, the aggregation operator (merging in this case) is believed to be uniquely defined by the context.

As shown in Table 23, there exist statistically significant differences in three quality attributes: 'Inconsistencies', 'Completeness' and 'Lessens Understandability'. The Cliff's Delta value shown in Table 24 indicates that the statistical significances observed in these three mentioned categories are in favor of the SSUCD subjects. The sensitivity analysis performed did not reveal any further significant information (Table 25).

Aggregation of Total Performance

Here we merge all quality attributes into a single meta-statement. The results are shown in Table 26. As shown in Table 27, there exists a statistically significant difference between the performances of SSUCD subjects in comparison to their UNL counterpart with respect to the overall quality achieved. As stated by the effect size test (Table 28), SSUCD subjects performed better overall that UNL subjects. The sensitivity analysis (Table 29) and uncertainty analysis performed (Table 30) did not reveal any further significant information.

TABLE 22. DESCRIPTIVE STATISTICS OF THE RESULTS

Quality Attribute	Technique	n	Median	IQR	95% CI of Median
Inconsistencies	**SSUCD**	33	1.000	0.400	0.800 to 1.000
	UNL	34	0.714	0.286	0.571 to 0.857
Completeness	**SSUCD**	33	0.909	0.143	0.905 to 1.000
	UNL	34	0.818	0.192	0.727 to 0.909
Fault-Free	**SSUCD**	33	0.875	0.375	0.625 to 0.875
	UNL	34	0.750	0.250	0.625 to 0.875
Analytical	**SSUCD**	33	1.000	0.333	1.000 to 1.000
	UNL	34	1.000	0.250	0.750 to 1.000
Understandability	**SSUCD**	33	0.750	0.375	0.625 to 0.875
	UNL	34	0.429	0.429	0.286 to 0.571

TABLE 24.
CLIFF'S DELTA FOR ALL QUALITY ATTRIBUTES

Quality Attribute	Cliff's delta ($\hat{\delta}$)	Variance	CI around delta ($\hat{\delta}$)	
			max	min
Inconsistencies	0.370	0.019	0.602	0.080
Completeness	0.352	0.019	0.588	0.060
Fault-Free	0.084	0.017	0.324	-0.167
Analytical	0.322	0.017	0.547	0.053
Understandability	0.467	0.018	0.685	0.173

TABLE 25. SENSITIVITY ANALYSIS OF THE RESULTS FOR ALL ATTRIBUTES

Quality Attribute	Original P value with all pairs considered	Change in P value after pairs removal	# of pairs removed	In favor of
Inconsistencies	0.008	Insignificant	3	UNL
Completeness	0.013	Insignificant	2	UNL
Fault-Free	0.557	Significant	6	UNL
		Significant	4	SSUCD
Analytical	0.480	Significant	10	UNL
		Significant	4	SSUCD
Understandability	0.001	Insignificant	5	UNL

TABLE 23. MANN-WHITNEY TEST FOR ALL QUALITY ATTRIBUTES

Quality Attribute	Technique	Rank sum	Mean rank	U	Difference between medians	95.2% CI	Mann-Whitney U statistic	2-tailed p
Inconsistencies	SSUCD	1329.5	40.29	353.5	0.143	0.000 to 0.286	353.5	0.008
	UNL	948.5	27.90	768.5				
Completeness	SSUCD	1319.5	39.98	363.5	0.091	0.000 to 0.177	363.5	0.013
	UNL	958.5	28.19	758.5				
Fault-Free	SSUCD	1168.0	35.39	515.0	0.000	-0.125 to 0.125	515	0.557
	UNL	1110.0	32.65	607.0				
Analytical	SSUCD	1169.5	35.44	513.5	0.000	0.000 to 0	513.5	0.480
	UNL	1108.5	32.60	608.5				
Understandability	Airline	1384.0	41.94	299.0	0.241	0.107 to 0.429	299	0.001
	Banking	894.0	26.29	823.0				

TABLE 27. MANN-WHITNEY TEST FOR ALL QUALITY ATTRIBUTES

Group	Rank sum	Mean rank	U	Difference between medians	95.2% CI	Mann-Whitney U statistic	2-tailed p
SSUCD	1360.0	41.210	323.0	0.562	0.185 to 0.917	323	0.0028
UNL	918.0	27.0	799.0				

TABLE 26. DESCRIPTIVE STATISTICS OF THE RESULTS

Group	n	Median	IQR	95% CI of Median
SSUCD	34	4.111	0.717	3.752 to 4.357
UNL	33	3.399	1.064	3.089 to 3.997

TABLE 28. CLIFF'S DELTA

Cliff's delta ($\hat{\delta}$)	Variance	Confidence Interval around delta ($\hat{\delta}$)	
		Max.	Min.
0.516	0.016	0.719	0.233

TABLE 29. SENSITIVITY ANALYSIS OF THE RESULTS

	Original P value before pairs removal	Change in P value after pairs removal	# of pairs removed	In favor of
Overall Quality	0.0028	Insignificant	2	UNL

TABLE 30. UNCERTAINTY ANALYSIS OF THE RESULTS

Quality Attribute	Assigned Weight	P value
Inconsistencies	0.1 ↔ 10	Significant
Correctness	0.1 ↔ 10	Significant
Incorrectness	0.1 ↔ 2	Significant
	3 ↔ 10	Insignificant
-Analytical	0.1 ↔ 7	Significant
	8 ↔ 10	Insignificant
Understandability	0.1 ↔ 10	Significant

VI. CONCLUSION

This paper argues that current analysis approaches in many empirical software engineering papers fail to fully explore their data sets. To achieve this additional exploration and insight, the field needs to be willing to embrace the use of composite metrics even within traditional hypothesis testing. While composite metrics have the ability to provide additional insight, they are not without risk. Hence, the paper argues that when composite metrics are utilized, the researcher needs to follow a careful process including components, which seek to illustrate the level of bias or uncertainty introduced by these composite measures. The paper undertakes a case study to demonstrate these ideas in practice. The paper specifically recommends that experimenters need to use both sensitivity and uncertainty analysis when utilizing composite measures. It is believed that the use of sensitivity and uncertainty analysis in combination is a novel contribution to the empirical software engineering literature and that treatise on the "correct" use of composite metrics provides an additional augmentation to existing practice.

APPENDIX A RAW DATA

TABLE 31. RAW DATA FOR AIRLINE TICKETING SYSTEM VS. BANKING SYSTEM RESULTS (INCONSISTENCIES)

Airline System		Banking System	
Inconsistencies		*Inconsistencies*	
SSUCD	UNL	SSUCD	UNL
-2	-3	0	-7
0	-5	0	-2
-1	-1	0	-6
0	-2	0	-2
0	-5	0	0
-5	-2	0	0
0	-6	-4	-1
0	-1	0	-3
-2	0	-3	-3
0	0	-2	-1
-2	-1	0	-1
-3	-6	-1	-3
-1	-3	-1	-3
-1	-5	0	-3
-1	-1	0	-1
-2	-6	0	0
0	-1		0

TABLE 32. RAW DATA FOR AIRLINE TICKETING SYSTEM VS. BANKING SYSTEM RESULTS (COMPLETENESS)

Airline System		Banking System	
Completeness (max. 11)		*Completeness (max. 21)*	
SSUCD	**UNL**	**SSUCD**	**UNL**
8	7	14	12
7	7	21	20
10	7	20	19
11	10	14	11
11	10	18	17
11	9	19	9
10	8	18	12
10	11	19	19
11	10	20	18
10	10	17	17
10	8	21	20
9	11	21	9
8	11	16	20
11	9	12	21
11	9	19	16
11	8	19	20
10	8		15

TABLE 33. RAW DATA FOR AIRLINE TICKETING SYSTEM VS. BANKING SYSTEM RESULTS (ANALYTICAL)

Airline System		Banking System	
Analytical		*Analytical*	
SSUCD	**UNL**	**SSUCD**	**UNL**
0	0	0	-1
0	-1	0	0
0	0	-1	-1
0	0	-1	0
-1	0	0	-1
0	0	-1	0
-1	-2	0	-1
-1	-1	-3	0
0	0	0	0
-1	-1	0	-1
0	0	0	0
0	-2	0	0
-1	0	0	0
0	-1	0	0
0	-2	0	0
0	0	0	-4
0	-1		0

TABLE 34. RAW DATA FOR AIRLINE TICKETING SYSTEM VS. BANKING SYSTEM RESULTS (FAULT-FREE)

Airline System		Banking System	
Fault-Free		*Fault-Free*	
SSUCD	**UNL**	**SSUCD**	**UNL**
-1	0	-3	-5
-1	0	-3	-2
-1	-2	-1	-1
-1	0	0	-1
-4	-3	0	-3
-2	-2	-8	-1
0	-4	-7	-2
0	0	-3	-1
-1	0	-2	-5
-4	-3	-7	-8
-1	0	-6	0
0	-8	-1	-3
0	0	-3	-5
-1	-5	-1	-3
0	-2	-4	-1
0	-8	-5	-1
0	-2		-1

TABLE 35.RAW DATA FOR AIRLINE TICKETING SYSTEM VS. BANKING SYSTEM RESULTS (UNDERSTANDABILITY)

Airline System		Banking System	
Understandability		*Understandability*	
SSUCD	**UNL**	**SSUCD**	**UNL**
0	-2	-5	-6
0	-3	-4	-4
-2	-5	-1	-4
-2	-3	-2	-7
-1	-6	-4	-5
-2	-2	-6	-6
-4	-3	-3	-5
-1	-2	0	-4
-1	-2	-3	-6
-1	-1	-6	-5
-1	0	-8	-3
-3	-6	-2	-1
-1	-3	-6	-4
-1	-5	-2	-6
-1	-3	-5	-6
-1	-4	-6	0
0	-1		-7

ACKNOWLEDGMENT

The authors would like to acknowledge the support provided by the Deanship of Scientific Research (DSR) at King Fahd University of Petroleum and Minerals (KFUPM) for funding this work through project No. IN111028.

REFERENCES

[1] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr, Basic Concepts and Taxonomy of Dependable and Secure Computing, IEEE Transactions on Dependable and Secure Computing, vol. 1, pp. 11-33, 2004.

[2] P. Beatty, I. Reay, S. Dick, and J. Miller, Consumer Trust in E-Commerce Websites: A Meta-Study, *ACM Computing Surveys*, (In print).

[3] R. Chillarege. Orthogonal Defect Classification. In M. R. Lyu, editor, Handbook of Software Reliability Engineering, chapter 9. IEEE Computer Society Press and McGraw-Hill, 1996.

[4] N. Cliff, Dominance statistics: Ordinal analyses To Answer Ordinal Questions, Psychological Bulletin, vol. 114, pp. 494-509, 1993.

[5] N. Cliff, "Answering Ordinal Questions With Ordinal Data Using Ordinal Statistics," Multivariate Behavioral Research, vol. 31, pp. 331-350, 1996.

[6] N. Cliff, *Ordinal Methods for Behavioral Data Analysis*. Lawrence Erlbaum Associates, 1996.

[7] H. W. Coleman and W.G. Steele, *Experimentation and Uncertainty Analysis for Engineers, 2nd Edition*. John Wiley & Sons, Inc., 1999.

[8] M. El Attar, J. Miller, Producing Robust Use Case Diagrams via Reverse Engineering of Use Case Descriptions, Journal of Software and Systems Modelling, vol. 7, No. 1, pp. 67 - 84, 2008.

[9] M. El-Attar, J. Miller, A Subject-Based Empirical Evaluation of SSUCD's Performance in Reducing Inconsistencies in Use Case Models, Journal of Empirical Software Engineering, Vol. 14, pp. 477 – 512, 2009.

[10] G. Gigerenzer, Z. Swijtink, T. Potter, L. Daston, J. Beatty, and L. Kruger, *The Empire of Chance: How Probability Changed Science and Everyday Life*. Cambridge University Press, 1989.

[11] H. Gomaa, *Designing Software Product Lines with UML*. Addison Wiley Professional, 2004.
[12] R. B. Grady. Practical Software Metrics for Project Management and Process Improvement. Prentice-Hall, 1992.
[13] G. Munda, Social Multi-Criteria Evaluation: Methodological Foundations and Operational Consequences, *European Journal of Operational Research*, vol. 158, pp. 662- 677, 2004.
[14] M. Nardo, M. Saisana, A. Saltelli, and S. Tarantola, "Tools for Composite Indicators Building," Report EUR 20408 EN, European Commission-Joint Research Centre, Institute for the Protection and Security of the Citizen, Econometrics and Statistical Support to Antifraud Unit, 2005.
[15] J. Neyman and Egon Pearson, On the Problem of the Most Efficient Tests of Statistical Hypotheses, *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, vol 231, pp. 289–337, 1933.
[16] T. E. Nichols and A. H. Holmes, Nonparametric Permutation Tests For Functional Neuroimaging: A Primer with Examples, *Human Brain Mapping*, vol. 15, pp. 1-25, 2001.
[17] G. Overgraad and K. Palmkvist, *Use Cases Patterns and Blueprints*. Addison-Wesley, 2005.
[18] M. Saisana and S. Tarantola, "State-Of-The-Art Report On Current Methodologies And Practices For Composite Indicator Development," Report EUR 20408 EN. European Commission-Joint Research Centre, Institute for the Protection and Security of the Citizen, Econometrics and Statistical Support to Antifraud Unit, 2002.
[19] M. Saisana, A. Saltelli, and S. Tarantola, Uncertainty and sensitivity analysis techniques as tools for quality assessment of composite indicators, Journal of the Royal Statistical Society. vol. 168, pp. 307–323, 2005.
[20] A. Saltelli, K. Chan, M. Scott, (eds) *Sensitivity Analysis: Gauging the Worth of Scientific Models*. Probability and Statistical Series, John Wiley and Sons, 2000.
[21] S. Siegel, and N. J. Castellan Jr., *Non-parametric Statistics for the Behavioral Sciences (2nd Edition)*. McGraw-Hill, 1988.
[22] J.L. Simon, *Resampling: The New Statistics*. Resampling Stats, 1995.
[23] E. H. Simpson, The Interpretation of Interaction in Contingency Tables, Journal of the Royal Statistical Society, Ser. B, vol. 13, pp. 238-241, 1951.

Mohamed El-Attar Dr. Mohamed El-Attar received his B.Eng. degree from Carleton University, Canada, and his Ph.D. degree from the University of Alberta, Canada. In 2009, he joined the department of Information and Computer Science at King Fahd University of Petroleum and Minerals, Saudi Arabia, as an assistant professor. His research interests include Requirements Engineering, in particular with UML and use cases, object-oriented analysis and design, model transformation and empirical studies. For information about his research see http://faculty.kfupm.edu.sa/ICS/melattar/index.html. Contact him at melattar@kdupm.edu.sa

James Miller received the B.Sc. and Ph.D. degrees in Computer Science from the University of Strathclyde, Scotland. During this period, he worked on the ESPRIT project GENEDIS on the production of a real-time stereovision system. Subsequently, he worked at the United Kingdom's National Electronic Research Initiative on Pattern Recognition as a Principal Scientist, before returning to the University of Strathclyde to accept a lectureship, and subsequently a senior lectureship in Computer Science. Initially during this period his research interests were in Computer Vision, and he was a co-investigator on the ESPRIT 2 project VIDIMUS. Since 1993, his research interests have been in Software and Systems Engineering. In 2000, he joined the Department of Electrical and Computer Engineering at the University of Alberta as a full professor and in 2003 became an adjunct professor at the Department of Electrical and Computer Engineering at the University of Calgary. He is the principal investigator in a number of research projects that investigate software verification, validation and evaluation issues across various domains, including embedded, web-based and ubiquitous environments. He has published over one hundred refereed journal and conference papers on Software and Systems Engineering (see www.steam.ualberta.ca for details on recent directions); and recently served as the program co-chair for the IEEE International Symposium on Empirical Software Engineering and Measurement; and sits on the editorial board of the Journal of Empirical Software Engineering. He regularly appears in the Journal of Systems and Software survey of "top scholars". This survey ranks leading researchers by their output in leading journals over a 5-year period. In the most recent survey, he was ranked the ninth most productive researcher in the world.

Call for Papers and Special Issues

Aims and Scope.

Journal of Software (JSW, ISSN 1796-217X) is a scholarly peer-reviewed international scientific journal focusing on theories, methods, and applications in software. It provide a high profile, leading edge forum for academic researchers, industrial professionals, engineers, consultants, managers, educators and policy makers working in the field to contribute and disseminate innovative new work on software.

We are interested in well-defined theoretical results and empirical studies that have potential impact on the construction, analysis, or management of software. The scope of this Journal ranges from the mechanisms through the development of principles to the application of those principles to specific environments. JSW invites original, previously unpublished, research, survey and tutorial papers, plus case studies and short research notes, on both applied and theoretical aspects of software. Topics of interest include, but are not restricted to:

- Software Requirements Engineering, Architectures and Design, Development and Maintenance, Project Management,
- Software Testing, Diagnosis, and Validation, Software Analysis, Assessment, and Evaluation, Theory and Formal Methods
- Design and Analysis of Algorithms, Human-Computer Interaction, Software Processes and Workflows
- Reverse Engineering and Software Maintenance, Aspect-Orientation and Feature Interaction, Object-Oriented Technology
- Component-Based Software Engineering, Computer-Supported Cooperative Work, Agent-Based Software Systems, Middleware Techniques
- AI and Knowledge Based Software Engineering, Empirical Software Engineering and Metrics
- Software Security, Safety and Reliability, Distribution and Parallelism, Databases
- Software Economics, Policy and Ethics, Tools and Development Environments, Programming Languages and Software Engineering
- Mobile and Ubiquitous Computing, Embedded and Real-time Software, Database, Data Mining, and Data Warehousing
- Internet and Information Systems Development, Web-Based Tools, Systems, and Environments, State-Of-The-Art Survey

Special Issue Guidelines

Special issues feature specifically aimed and targeted topics of interest contributed by authors responding to a particular Call for Papers or by invitation, edited by guest editor(s). We encourage you to submit proposals for creating special issues in areas that are of interest to the Journal. Preference will be given to proposals that cover some unique aspect of the technology and ones that include subjects that are timely and useful to the readers of the Journal. A Special Issue is typically made of 10 to 15 papers, with each paper 8 to 12 pages of length.

The following information should be included as part of the proposal:

- Proposed title for the Special Issue
- Description of the topic area to be focused upon and justification
- Review process for the selection and rejection of papers.
- Name, contact, position, affiliation, and biography of the Guest Editor(s)
- List of potential reviewers
- Potential authors to the issue
- Tentative time-table for the call for papers and reviews

If a proposal is accepted, the guest editor will be responsible for:

- Preparing the “Call for Papers” to be included on the Journal’s Web site.
- Distribution of the Call for Papers broadly to various mailing lists and sites.
- Getting submissions, arranging review process, making decisions, and carrying out all correspondence with the authors. Authors should be informed the Instructions for Authors.
- Providing us the completed and approved final versions of the papers formatted in the Journal’s style, together with all authors’ contact information.
- Writing a one- or two-page introductory editorial to be published in the Special Issue.

Special Issue for a Conference/Workshop

A special issue for a Conference/Workshop is usually released in association with the committee members of the Conference/Workshop like general chairs and/or program chairs who are appointed as the Guest Editors of the Special Issue. Special Issue for a Conference/Workshop is typically made of 10 to 15 papers, with each paper 8 to 12 pages of length.

Guest Editors are involved in the following steps in guest-editing a Special Issue based on a Conference/Workshop:

- Selecting a Title for the Special Issue, e.g. “Special Issue: Selected Best Papers of XYZ Conference”.
- Sending us a formal “Letter of Intent” for the Special Issue.
- Creating a “Call for Papers” for the Special Issue, posting it on the conference web site, and publicizing it to the conference attendees. Information about the Journal and Academy Publisher can be included in the Call for Papers.
- Establishing criteria for paper selection/rejections. The papers can be nominated based on multiple criteria, e.g. rank in review process plus the evaluation from the Session Chairs and the feedback from the Conference attendees.
- Selecting and inviting submissions, arranging review process, making decisions, and carrying out all correspondence with the authors. Authors should be informed the Author Instructions. Usually, the Proceedings manuscripts should be expanded and enhanced.
- Providing us the completed and approved final versions of the papers formatted in the Journal’s style, together with all authors’ contact information.
- Writing a one- or two-page introductory editorial to be published in the Special Issue.

More information is available on the web site at http://www.academypublisher.com/jsw/.

(Contents Continued from Back Cover)

REGULAR PAPERS