Predicting Software Damage Rate from Cognitive Bias in Software Design Process

NoptanitChotisarn,NakornthipPrompoon

Software Engineering Laboratory, Center of Excellence for Software Engineering
Department of Computer Engineering, Faculty of Engineering
Chulalongkorn University, Bangkok, Thailand
Noptanit.C@Student.chula.ac.th, Nakornthip.S@chula.ac.th

Abstract

Software design process is one of the critical processes in software development life cycle since it delivers a software design specification basically used for implementation in order to deliver software meets user expectation. There are many tasks required different skills of design team members. Due to the different backgrounds, experiences and preferences of software designers, cognitive bias may arise that leads tothe increase of software damage. In this paper, we want to explore whether there is a correlation between any cognitive bias types occur from human working process and software damage rate. From our experiment, we discover that there is a statistically significantcorrelation between the two types of cognitive bias; confirmation bias and adjustment bias in design process and software damage rate.

Keywords: Human Factor, Software Psychology, Software Design, Cognitive Bias, Software Requirements Specification

1. Introduction

In software development for a large scale project, there are usually composed of different processes occupied by different skilled people with different backgrounds. From [3], there were three types of bias may effect on software development. They are listed as follows.

Anchoring and adjustment; anchoring bias is the initial estimate of solution for any problem of human,adjustment bias is to modify those initial estimates for getting a revised final solution is an appropriate way [5], [6].

Availability; this bias suggests that human is influenced by information that is easy to recall from dissemination of information in the media or event just happened recently [5].

Confirmation; this bias suggests that people tend to focus on the information that is consistent with the concept that they previously thought whereas they ignore information that is inconsistent with the concept of their past [7].

According to ourprevious research [11] which focuses on the four biases;anchoring, adjustment, availability, and confirmation, we wanted to study whether one of them has an effect on software damage rate in software requirements gathering and specification process. We found out thatonly adjustment, availability, and confirmation bias had an effect on software damage rate. In addition, there is a strong collinearity between anchoring bias and adjustment bias.

This our previous research result motivated us to study more on the cognitive bias and software damage rate in the design process. In the design process of a complex software development, there are

various skills needed from software designer team. Each software designer may have different backgrounds, experiences and preferences. Cognitive bias may occur in any steps of design processes. We would like to study whether, once there is a bias occurs in the design work process, it will negatively affect the quality of the software which we will focus on the software damage rate. Software damage rate refers to the defect of software after deploy to the real operating environment.

This research will study two aspects. First, we would like tostudy whether there isat least one of the cognitive biases affect the quality of the software product during software design process. Second, we would like to study whether anchoring bias and adjustment bias which are highly related in our previous research are still statistically collinearity.

2. Related Researches

There is researched studied in aspect of software engineering, focusing on how to best associate personnel with the various tasks in a software project to determine who are best suited to particular development roles and to emphasize the important of skill diversity in the software engineering field. This may consequently introduce bias from human during software development [4]. People make biased decisions. Especially, software designers may also be subjected to such cognitive biases which may lead to unsound design decisions [1].

There is provided the two stages information model that illustrated there error and bias might occur during the transformation from raw data to the project required documentation [2].Thus, basically based on this research, we would like to further more on the error comes from bias on design phase which also has the similar data transformation requirements. The various types of biases may effect on

software development. There were four types of bias; anchoring, adjustment, availability, and confirmation; usually occur in software engineering aspect [3].Thus, these four biases were selected to study in our research.An analyzed the independent variables with the dependent variable by using multiple regressions and correlation analysis to find factors might affect the cost of software customization [8].We would like to apply the same statistical method to our research. They have proven that there were three types of bias might occur during requirement gathering and specification process and there were a strong collinearity between anchoring bias and adjustment bias [11].

3. Related Principles

3.1. Cognitive Bias

The cognitive bias is a pattern of tendency occurrence in decision making in various kinds of situations. Cognitive bias may cause a distorted perception of the decision which leads to the incorrectness and unreasonable interpretation is also called irrationality [9], [10]. Cognitive bias is a part of the thinking and decision making. The most popular cognitive biases forvarious solutions are anchoring and adjustment, availability, and confirmation bias [3], that we had described those biases in introduction part.

3.2. Two-stage Information Model

A two-stage information model present how the data is changed from one step to another by one is responsible for a particular stage. This model has been used in the research of Andrew P. Snow and Mark Keil [2] in the status report preparation which transforms project status data to status report document. This research present that there are error and bias usually occur in this process. In addition, our

previous research [11] also applied the two-stage status model in the requirements gathering and specification process.

4. Research Methods

To evaluate our motivation, this section describes the experimental design. The experiment is started from selecting cognitive bias types, and setting research hypothesis, defining target samples unit and surveying using questionnaire. Then, the collected information is analyzed using regression and correlation analysis. Finally, the result found is presented.

4.1. Cognitive Bias Selection

This research is a continue research from the previous one [11]. We use the same cognitive bias types. They are adjustment bias, availability bias, and confirmation bias since all are usually find in software engineering [3].

4.2. Setting the Hypothesis

The hypothesis of the research based on multiple regressions and correlation analysis methods is present as following:

H0: Independent variables are the cognitive biases factors that have no effect on the software damage rate in terms of software design process.

H1: Independent variables are the cognitive biases factors that influence on the software damage rate in terms of software design process.

If H1 is accepted, we will analyze the result that there is the relationship of anchoring bias and adjustment bias or not.

4.3. Sample and Survey

Our research conducts by using 30 samples from graduated students in computer or management field who have experience more than one year in computer field related to software design process.

The survey method using set of ques-

tion was used to collect the information about the quantity of occurrences of bias, by taking the keyword properties of various biases in the situation of the software design process. Also the sample unit would identify a quantity of software damage rate after the software product has been launched according to each type of the focus biases.

The sample of keyword of each type of biases; anchoring, adjustment, availability and confirmation are "experience", "customization", "modernity" and "confidence" respectively [11].

5. Results

This section discusses the statistical computing results are the resulting from the number of errors that occur in the software product. To define Y is the software damage rate from occurrence of cognitive bias in software design process. To defines X is occurrence of cognitive bias situation in software design process by using multiple regressions and correlation analysis and stepwise.

5.1. Define X and yAVG1 variables

The dependent variable or yAVG1 is defined by using the average values of software damage rate from each bias situation (anchoring, adjustment, availability, and confirmation bias) as y1, y2, y3, and y4 are shown in table I.

The X value is an independent variable that gathering from a survey. Value of X means the value number of situations that has biased in 10 times of the working process: anchoring, adjustment, availability, and confirmation as x1, x2, x3, and x4 respectively.

TABLE I EXAMPLE OF AVERAGE VALUE OF
SOFTWARE DAMAGE RATE

Sample	y1	y2	y3	y4	yAVG1
1	0	2	1	0	0.75
2	3	8	8	7	6.50
3	0	0	0	0	0.00
…	…	…	…	…	…
30	1	3	1	1	1.50

5.2. Statistical Analysis and Results

Using statistical application for linear regression analysis, first we need to set yAVG1 to the dependent variable and set x1, x2, x3, and x4 as independent variables.Then we selects the function for calculating the regression coefficient, model fit of linear regression, standard error of estimate (SEE) and selecting the function to plotthe histogram and normal probability. We can further to next step, if the data distribution is a normal curve. From our survey experiment, the data is in a normal distribution.The relationship result between variables is presented in table II.

TABLE II   VARIABLES ENTERED/REMOVED[A,B]

Model	Variables Entered	Variables Removed
1	x4	-
2	x2	-

a. Dependent Variable: yAVG1
b. Stepwise(Criteria:Probability-of-F-to-enter <= 0.050,Probability-of-F-to-remove >= .100).

For the result as shown in table II, the stepwise criteria is used for choosing independent variables; each type of cognitive bias variables. Only x4 (confirmation bias) and x2 (adjustment bias)are selected to form the regression equation. This means that confirmation bias and adjustment are related to the software damage rate. The strength level of correlation is shown in Table III.

TABLE III   CORRELATIONS I

		yAVG1	x2	x4
Pearson Correlation	yAVG1	1.000	0.638	0.795
	x2	0.638	1.000	0.554
	x4	0.795	0.554	1.000

From table III, we can also see that x4 has a higher relationship toyAVG1 than x2. However, both biases are correlated to software damage rate with the value greater than 0.500 and closely to 1.000 that is similar to table II.

In addition, anchoring and adjustment bias were highly collinearity since the computed result from a 1-tailed test is 0.0 as shown in Table IV. This result also confirms the result from the previous work [11]. Thus, we can imply that both biases always happen together in the same event of software damage rate.

TABLE IV   CORRELATIONS II

		yAVG1	x1	x2
Sig. (1-tailed)	yAVG1	.	0.000	0.000
	x1	0.000	.	0.000
	x2	0.000	0.000	.

Our result is also confirmed from histogram and normal p-plot regression standard residual. Both represents that the data is a normal distribution and tends to be linear.
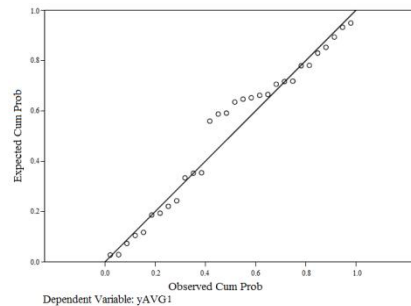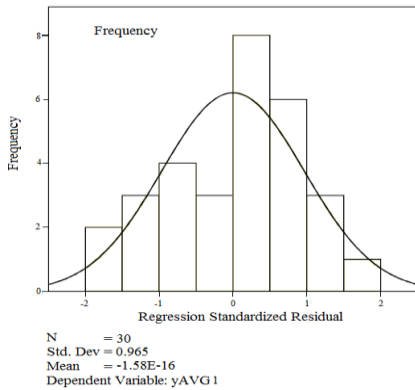

Fig. 1: Normal p-plot regression standard residual

Fig. 2: Histogram of normal distribution

6. Conclusion and Future Work

According to the experiment results, H1 is accepted that is independent variables; confirmation bias and adjustment bias have an influence on the software damage rate in terms of software design process. In addition, there is a highly collinearitybetween anchoring and adjustment bias.

The experiment result could apply to the software project management to decrease cognitive bias in software design process in order to reduce software damage rate.Thus, project manager should concern the bias issue by investigating the feasible solutions to lessen the candidate factor may effect to bias occurrence.

Our future works will move on to the risk management aspect that arises from the cognitive bias.

7. References

[1] Antony Tang, "Software Designers, Are You Biased?," ACM, 2011.

[2] Andrew P. Snow, Mark Keil, and Linda Wallace, "The effects of optimistic and pessimistic biasing on software project status reporting," Information & Management 44, pp. 130–141, 2007.

[3] Kannan Mohan, and Radhika Jain, "Using Traceability to Mitigate Cognitive Biases in Software Development," Communications of The ACM, vol. 51, no. 9, pp. 110-114, 2008.

[4] Luiz Fernando Capretz, and Faheem Ahmed, "Why Do We Need Personality Diversity in Software Engineering?," ACM Sigsoft Software Engineering Notes, vol. 35, no. 2, pp. 1-11, 2010.

[5] Stacy W., and MacMillan J., "Cognitive bias in software engineering," Communications of the ACM, 38 (6), pp. 57 – 63, 1995.

[6] Parsons J., and Saunders C., "Cognitive Heuristics in Software Engineering: Applying and Extending Anchoring and Adjustment to Artifact Reuse," IEEE Transactions on Software Engineering, 30 (12), pp. 873-888, 2004.

[7] Fisher K., and Statman M., "Cognitive Biases in Market Forecasts: The frailty of forecasting," The Journal of Portfolio Management, pp. 1-10, 2000.

[8] Hathaichanok Suwanjang., and Nakornthip Prompoon., "Framework for Developing a Software Cost Estimation Model for Software Modification Based on a Relational Matrix of Project Profile and Software Cost Using and Analogy Estimation Method," ICSIE 2012, 2012.

[9] Baron J., Thinking and deciding (4th ed.), New York, NY: Cambridge University Press, 2007.

[10] Ariely D., "Predictably irrational: The hidden forces that shape our decisions," New York, NY: HarperCollins, 2008.

[11] Noptanit Chotisarn., and Nakornthip Prompoon., "Forecasting Software Damage Rate from Cognitive Bias in Software Requirements Gathering and Specification Process," ICIST 2013, to be published.