Pairwise Programming Software Development Reliability Prediction by using GAM

Somchai Prakancharoen
Department of computer and information science, Faculty of applied science
King Mongkut's university of technology north bangkok
CountryBangkok, Thailand
spk@kmutnb.ac.th

บทคัดย่อ—บทความนี้มีวัตถุประสงค์ในการพัฒนาวิธีการในการพิจารณาการมอบหมายให้นักพัฒนาโปรแกรมสองคนมาทำการพัฒนาโปรแกรมหนึ่งๆร่วมกันเพื่อให้โปรแกรมที่ได้มีความผิดพลาดน้อยที่สุด ค่าสมรรถนะของแต่ละคนถูกนำมาใช้ในการสร้างสมการถดถอยเพื่อประมาณการความผิดพลาดของซอฟต์แวร์ที่พัฒนาได้และเมื่อนักพัฒนาโปรแกรมสองคนมาทำงานร่วมกันจะพิจารณาคุณลักษณะของตัวบุคคล ว่าตรงกับลักษณะของซอฟต์แวร์ที่ต้องการพัฒนามากน้อยเพียงใด จากนั้นทำการสร้างสมการถดถอยที่ทำการพยากรณ์ค่าความผิดพลาดของโปรแกรม การพัฒนาสมการพยากรณ์ในการพยากรณ์แบบจำลองเพิ่มจากสมการทั่วไป จะกระทำจากโครงการพัฒนาซอฟต์แวร์เดิมรวมสามสิบโครงการ แบบจำลองเพิ่มจัดสร้างจากสมการถดถอยทั้งสองดังกล่าว พบว่าสมการมีความสามารถในการอธิบายข้อมูลได้ที่ R^2=0.616 และมีค่าความถูกต้องในการพยากรณ์ความผิดพลาดของโปรแกรมเท่ากับ 78.70%

คำสำคัญ: การพัฒนาโปรแกรมแบบคู่นักพัฒนาโปรแกรม, แบบจำลองเพิ่มจากสมการทั่วไป

Abstract—**The objective of this research was to construct a method of choosing right two programmers to work together in pairwise programming in order to decrease software development error rate. Independence variable were programmer's competency and personal type which were considered of suitable with particular software requirement. Thirty finished software projects, pairwise software development were collected then GAM technique was used to construct a prediction software project error rate. R^2 of GAM equation was 0.616. Accuracy test of prediction was 78.70%**

Keywords- Pairwise programming, Generalized additive model

I. INTRODUCTION

Pairwise programming software development is a practical method of increasing the software quality, cross checking, productivity etc.

Most of software project have their specific requirements and constraints. If software project manager could select a right couple of programmers that match to project's requirement and constraints then software project might be success in purpose.

This research suggests a solution of choosing a right two programmers to work in a particular distinct software project. Experiment observations were gathered from thirty finished pair wise software development projects. Prediction equation was constructed from all thirty observations with generalized additive model. Project' software error rate is a dependent variable that project manager want to know in order to select the right two programmers. Prediction equation fitting to experiment observation was explained in R^2.

II.RELATED THEORY AND RELATED RESEARCH

A. Independence and dependent variable

Interested independence variable that should predict software development error rate were gathered from many research [1, 2, 3] then these variables were chosen by three experts in three Thailand software houses.

- Independence variable

There are five competencies and one personal type of each programmer.

Programmer#1

X1_1 = How relevance experience to assigned software project - RE (1-100%)
X1_2 = Relevance knowledge-RK(1-100%)
X1_3 = Task availability-TA (1-100%)
X1_4 = Human relationship -HR(1-100%)
X1_5 = Attention-AT (1-100%)
X1_6 = Personal type – PT (Professional oriented-Task oriented-Interaction oriented (Likerts score range 5-5-5))
[represented as X1_6_1 = PO, X1_6_2 =TO, X1_6_3 =IO]

Programmer# 2
X2_1 = How relevance experience to assigned software

project - RE (1-100%)
X2_2 = Relevance knowledge-RK(1-100%)
X2_3 = Task availability-TA (1-100%)
X2_4 = Human relationship -HR(1-100%)
X2_5 = Attention-AT (1-100%)
X2_6 = Personal type – PT (Professional oriented-Task oriented-Interaction oriented (Likerts score range 5-5-5))
[represented as X2_6_1 = PO, X2_6_2 =TO, X2_6_3 =IO]

- Dependent variable

Dependence of this prediction equation is error rate of each software project development.

Error rate, Y, is defined as percentage of software error rate, for example Y=10%.

$$Error\_rate(Y) = \frac{\#Line-of-error}{Total-line-of-sourcecode} \quad (1)$$

- For individual software project, there have difference in required personal type that should be preferred in the software project difference stage.

For example, the personal type of human resource that suitable in software requirement engineering stage were professional type =4 of 5, task oriented = 2 of 5 and interaction oriented= 3 of 5.

This proportion could be represented as (4:2:3).

The proportion of personal type is not the same amount in difference software development stage.

However, in the same software development stage under difference infrastructure should have difference personal type proportion.

It was based on software project manager consideration.

The required personal type variables (e.g. person#1&#2) were defined as followed.
X12_6_1=Software project's expected in PO
X12_6_2=Software project's expected in TO
X12_6_3=Software project's expected in IO.

B. Generalized additive model: GAM [9]

GAM is a technique that could combine many difference equations, $f_j(x)$, which all could predict the same dependence variable (y). Each $f_j(x)$ may have a difference kind of model such as linear regression, nonlinear regression etc.

$$Y_{estimate} = \beta_0 + \sum_{j=1}^{k} (\beta_j . f_j(x)); i = 1, n \quad (2)$$

While

$f_j(x)$; j=1,k are prediction equation for Y.

β_0 is an intercept to Y axis.

β_j:j=1,k are coefficients (weighting) of each $f_j(x)$.

n is an number of observations which should be equal or not in each $f_j(x)$.

Parameters of (1) should be calculated from least square error method as equation (3) and (4).

$$\varepsilon^2 = (Y - Y_{estimate})^2 \quad (3)$$

$$\frac{\partial\varepsilon^2}{\beta_0}, \frac{\partial\varepsilon^2}{\beta_1}, \ldots \frac{\partial\varepsilon^2}{\beta_k} \Rightarrow 0 \quad (4)$$

C. Mean magnitude relative error: MMRE

MRE is a formula of error measurement of estimated value difference from real value. If there are many case then average of MRE (MMRE) was shown in equation (5).

In order to simply understand, multiply MRE with 100 and subtract it from 100 then this result should represent accuracy of estimation in percentage.

$$MMRE = \frac{1}{n} . \sum_{i=1}^{n} \frac{|\widehat{Y}_i - Y_i|}{Y_i} \quad (5)$$

$$Accuracy = 100 - MMRE * 100 \quad (6)$$

***D.**-Related research*

[1] There are three essential fundamental persona competency of IT such as effectiveness competency, academic competency, workplace competency. [1] has identify that the knowledge, skills, and abilities are needed for workers to perform successfully in the field of IT.

[2] This paper suggest software reliability prediction model on secrecy software development project based on software metrics. Metrics measurement were considered on Defect density (DD), Test coverage (TC), Requirements traceability (RT), Function point analysis (FP) and Bugs per line of code (BLOC)

[3] Software reliability was indirectly measured through software metrics by four categories: Product metrics, Project management metrics, Process metrics and Fault and failure metrics.

III.RESEARCH METHODOLOGY AND RESULT

According to actually working, there are difference manners. Task may be assigned to only one programmer or group of programmers. Therefore in this research, task allocation should be considered in two accounts.

The first is that how to predict efficiency of task perform under working by individual programmer.

The second is that how to predict efficiency of task perform under working together by two programmers.

To solve the first question, regression analysis of each programmer competencies were calculated in order to predict Y, represented as $\hat{Y}_1$.

When two selected programmers were work together, their three personal types of both were calculated for average value of PO, TO and IO.

Then, these three average values were subtracted to project's defined PO, TO and IO respectively. These three results were summed together which was called as grouping personal type (XG_6_12).

To solve the second question, regression analysis of both programmer personal types were calculated in order to predict Y, represented as $\hat{Y}_2$.

A. *Experimental data*

Experiment observations were collected in two periods of time.

- The first data collection.

Task (Program) was assigned Individual programmer, which has his own competency.

Personal type of each one was not considered in this period since each programmer was work alone. The allocated task was prepared for single person to design, coding and testing until this program was correctly finished. At the finished time, percentage of software's error rate was computed. Percentage of software error rate was defined as in equation (8).

Thirty software tasks observation were collected, partial of them were shown in table 1.

TABLE I. Partial of Single Programmer Task Allocation with all Variables

x_1	x_2	x_3	x1_4	x_5	x_6_1	x_6_2	x_6_3	y
80	80	80	80	80	5	2	2	5
82	81	86	79	83	3	4	1	5
78	77	81	86	91	1	1	5	5
82	81	77	85	81	2	2	4	5
77	75	80	85	85	4	4	1	5
80	82	81	75	78	5	1	1	10
90	95	85	88	80	3	2	1	10
79	85	91	85	79	2	2	5	10

- Second data collection.

This data were collected from a finished software projects that have assigned two programmers to work together. All variables, competencies and personal type, were collected together with "Error rate" (Y) variable.

Thirty software tasks observation were collected, partial of them were shown in table 2.

TABLE II. Partial of Single Programmer Task Allocation with all Variables

x1_1	x1_2	x1_3	x1_4	x1_5	x1_6_1	x1_6_2	x1_6_3	x2_1	x2_2	x2_3	x2_4	x2_5	x2_6_1	x2_6_2	x2_6_3
88	80	88	85	88	5	2	2	85	80	88	85	88	5	1	2
90	81	86	83	83	3	4	2	92	80	86	83	83	4	3	3
85	77	87	86	91	3	3	1	85	77	87	86	91	4	5	2
85	81	87	88	81	5	5	3	86	81	87	87	87	5	5	5
85	78	85	84	85	5	5	2	85	77	74	84	85	4	5	2
78	77	77	77	77	3	1	3	80	77	77	77	77	4	5	3

The personal type variables were explained in table 3. For example, variable "x12_6_1" is calculated from (x1_6_1 + x2_6_1)/2. Variable "xp_6_1" is an expected weight of PO which is subjected to software project manager's thought. Variable "xg_6_12" is total mark which calculated from summation of all subtraction of each personal type with expected weighting of project as shown in (7).

$$XG\_6\_12 = \{(X12\_6\_1 - XP\_6\_1) + (X12\_6\_2 - XP\_6\_2) + (X12\_6\_3 - XP\_6\_3)\} + 15 \quad (7)$$

TABLE III. Personal Type Variable

Variable name	Definition	Datatype	Range
x1_6_1	Programmer#1 , Personaltype= Professional oriented	Order	Likert 1-5
x1_6_2	Programmer#1 , Personaltype=Task oriented	Order	Likert 1-5
x1_6_3	Programmer#1 , Personaltype=Interactionl oriented	Order	Likert 1-5
x2_6_1	Programmer#2 , Personaltype= Professional oriented	Order	Likert 1-5
x2_6_2	Programmer#2 , Personaltype=Taskl oriented	Order	Likert 1-5
x2_6_3	Programmer#2 , Personaltype= Interaction oriented	Order	Likert 1-5
x12_6_1	Programmer#1&2-average , Personaltype= Professional oriented	Order	Likert 1-5
x12_6_2	Programmer#1&2 average , Personaltype=Task oriented	Order	Likert 1-5
x12_6_3	Programmer#1&2 average , Personaltype=Interactionl oriented	Order	Likert 1-5
xp_6_1	Programmer#Project weight , Personaltype= Professional oriented	Order	Likert 1-5
xp_6_2	Programmer#Project weight , Personaltype=Task oriented	Order	Likert 1-5
xp_6_3	Programmer#Project weight, Personaltype=Interactionl oriented	Order	Likert 1-5
xg_6_12	Programmer#Project total mark, Personaltype=all	Continous	0-20

Data preparation of experiment observation which include expected weight and total marks were shown in table 4.

TABLE IV. Partial Data Preparation of Personal Type (when two programmers were work together)

x2_6_1	x2_6_2	x2_6_3	X12_6_1	X12_6_2	X12_6_3	XP_6_1	XP_6_2	XP_6_3	XG_6_12
5	1	2	5	1.5	2	5	1	3	14.5
4	3	3	3.5	3.5	2.5	5	2	3	14.5
4	5	2	3.5	4	1.5	3	4	1	16
5	5	5	5	5	4	5	1	3	20
4	5	2	4.5	5	2	3	4	1	18.5
4	5	3	3.5	3	3	4	2	4	14.5
4	4	2	3.5	2.5	1.5	5	4	2	11.5
5	4	2	5	4.5	1.5	4	5	1	16
5	5	3	5	5	2.5	3	2	3	19.5
3	5	2	2.5	5	2	2	5	2	15.5

B. *Individual programmer Regression analysis in "Error rate".*

The first equation to predict dependence variable error rate (Y) was calculated from 30 observations with x_1, x_2, x_3, x_4, x_5 of 30 observations whereas he work alone. Best fitted regression equation was in a linear form as equation (8).

$$\hat{Y}_1 = 41.606 - 0.864.X4 + 0.374.X5 \quad , R^2 = 0.360 \qquad (8)$$

When two programmers work together equation (8) must be prepared average value of significance independence variables as equation (9).

$$\hat{Y}_1 = 41.606 - 0.864.(X1\_4 + X2\_4)/2 + 0.374.(X1\_5 + X2\_5)/2 \qquad (9)$$

C. Pair wise programmer Regression analysis in "Error rate"

When two programmers work together in assigned task, regression equation (10) was calculated from independence variable xg_6_12 and dependence variable error rate (Y).

$$\widehat{Y}_2 = 27.004 - 1.078.XG\_6\_12\,,\; R^2 = 0.501 \qquad (10)$$

D. Pair wise programmer Generalized additive model in "error rate".

Equation (9) and (10) were presented in difference independence variables but both were predict error rate therefore GAM was derived by combine them together to be a one GAM regression as equation (11).

$$\widehat{Y}_{Gam} = -0.746 + 0.472.Y_1 + 0.836.Y_2\,,\; R^2 = 0.616 \qquad (11)$$

E. Evaluation

Ten finished software projects were used to perform cross validation of (11) by MMRE. MMRE value is 0.213 of error (or about 21.30%). Therefore (11) could predict error rate of software development correctly about 78.70%.

II. SUMMARY AND SUGGESTION

Independence variables used in this research were chosen by three software project leaders of Thailand software house.

Experiment software project were in field business client server architecture application.

According to restriction of research budget year, thirty sample software project were collected from past – finished software projects which were developed under pairwise programming technique.

For model usage, software project manager could use this model (eq. 11) to support his decision by choosing two programmers then fill in their competency in equation 9 and 10 in order to calculate for $\hat{Y}_1$ and $\hat{Y}_2$.

After that, $\hat{Y}_{Gam}$ could be easily obtained. There are many permuted pairs. The pair of programmer which gave least value $\hat{Y}_{Gam}$ (least error rate) was then easily chosen.

REFERENCES

[1] Information Technology Competency Model, U.S. Department of Labor, Employment and Training Administration, 2016 State of Minnesota, available at: http://www.careeronestop.org/competencymodel/competency-models/information-technology.aspx

[2] Ying Shi, Metric-based software reliability prediction approach and its application, Empirical Software Engineering, (2016). doi:10.1007/s10664-016-9425.

[3] Jiantao Pan, Software Reliability, Dependable Embedded Systems, Carnegie Mellon University,Spring 1999.

[4] Trevor hastie, Generalized additive model, statistical science , vol 1, 1986.