ISBN 978-981-11-3671-9
Proceedings of 2017 the 7th International Workshop on Computer Science and Engineering (WCSE 2017)
Beijing, 25-27 June, 2017, pp. 181-187
doi: 10.18178/wcse.2017.06.032

Redundancy Weighting for Software Effort Estimation with Case Based Reasoning

Qin Liu [1], Jiakai Xiao [2+] and Hongming Zhu [1]

[1]School of Software Engineering, Tongji University
[2] Department of Computer Science and Technology, Tongji University

Abstract. Case-based reasoning (CBR) is a widely used approach in software effort estimation (SEE). Unfortunately, it may be over fed by redundant feature(s) that may lead to erroneous prediction. To alleviate the problem, this paper proposes a Relevance-Redundancy (R2D) distance that incorporates redundancy weighting. Experiment results demonstrate that R2D achieves optimal MAR and Pred(25) on 4 benchmark datasets with an average improvement of 17.4% and 27.8% against second optimal methods.

Keywords: feature weighting, redundancy, mutual information, case based reasoning

1. Introduction

To estimate the software project effort, many approaches have been proposed which include expertise-based techniques, regression based techniques, model-based techniques like COCOMO, learning oriented techniques like case based estimation, dynamic-based techniques, and composite techniques. According to the survey by Boehm et al. [1], these Software Effort Estimation (SEE) techniques are useful in many aspects including budgeting, trade off and risk analysis, project planning and control, and software improvement investment analysis.

Among these techniques, Case Based Reasoning (CBR) is widely accepted due to its easy adoption and interpretability [2]. In the literature, many CBR approaches need to adapt a distance measure to identify similar project(s) in predicting the effort. However, most of the existing measures focus on the relevance between features and effort, but rarely consider the redundancy among features [3,4]. To further explore this problem, we propose a mutual information based Relevance-Redundancy (R2D) feature weighting method, which recasts the notion of distance measure in CBR. The Relevance and Redundancy in our proposed distance measure are defined as the following:

- Relevance: There are many definitions of relevance in the literature [5]. Generally, it describes the correlation between effort and features. We will follow the idea that the more relevance for a feature, the higher feature weight it would be assigned in the effort estimation model [6].
- Redundancy: Redundancy refers to the correlation among features [7]. The effort estimation model might be over fed by redundant features, which leads to bias for the prediction accuracy.

This work is motivated by the hypothesis that *the software effort estimation accuracy may be further improved by introducing redundancy weighting in CBR*. The R2D measure is proposed to testify the assumption.

The remainder of the paper is organized as follows. Section 2 introduces related work on feature weighting and definition of information measures. Section 3 presents the proposed R2D method. Section 4

+ Corresponding author. Tel.: +8613120988302
E-mail address:waitingxjk@126.com.

introduces the experiment settings and Section 5 presents the experimental results. Section 7 concludes the paper.

2. Related Work

2.1. Feature weighting

Feature Weighting Techniques (FWTs) are common practice to set feature weight in distance measures and they can help improve estimation accuracy [8,9]. There are two major kinds of FWTs, namely wrappers and filters [6]. Typical wrappers include genetic methods [10], Particle Swarm Optimization (PSO) based methods [3] and extensive search [11]. Wrappers are computationally demanding in a way that they apply CBR repeatedly with a searching strategy to search better feature weights during each iteration. Filters, on the other hand, use statistical measures [4,12] to measure the importance of features and they are less computationally demanding than wrappers [3]. Generally, FWTs are designed to assign weights to each individual feature. In this case, only feature relevance can be addressed since no correlation between features is accounted.

2.2. Information measures

In the proposed feature weighting method, we would recast the notion of distance by mutual information (I), normalized mutual information (NI) and conditional mutual information (CI). The definitions of these concepts are the same to discrete variables and continuous variables. All the following equations apply for continuous variables with the summations being replaced by integrals.

Let's denote $X_i = \langle x_{i1}, ..., x_{im} \rangle$ as a discrete feature with probability density function (*pdf*) $p(x) = Prob(X_i = x)$. In the same way, we have X_j and Y. The entropy $H(X_i)$ defined by Eq.1 is the amount of uncertainty (or information) embedded in X_i. Typically we will always assume $0 log 0 = 0$. Usually the base of the logarithm is 2, so information is measured in bits.

$$H(X_i) = -\sum_x p(x) log p(x) \tag{1}$$

The mutual information $I(X_i;Y)$ (Eq.2) measures the amount of information shared by X_i and Y and it can be used to measure the relevance of X_i to Y.

$$\begin{aligned} I(X_i;Y) &= -\sum_{x,y} p(x,y) log \frac{p(x,y)}{p(x)p(y)} \\ &= H(X_i) + H(Y) - H(X,Y) \end{aligned} \tag{2}$$

The normalized mutual information [13] provides a normalized version of mutual information, hence it can be used as a relevance measure. It is defined as follows:

$$NI(X_i,Y) = \frac{I(X_i;Y)}{min\{H(X_i), H(Y)\}} \tag{3}$$

The conditional mutual information measures the unknown part of mutual information between two variables. It is defined as the following:

$$CI(X_i; X_j | Y) = I(X_i; X_j, Y) - I(X_i; Y) \tag{4}$$

3. The Proposed Method

This section introduces the feature relevance and redundancy weighting methods adapted in this paper. Based on the weighting method, a measure called Relevance-Redundancy Distance (R2D) is proposed to integrate redundancy weighting with relevance weighting in the distance measure.

Before we present R2D, we need to define how to measure feature relevance $Rel(X_i,Y)$ and redundancy $Red(X_i, X_j)$. Feature relevance $Rel(X_i,Y)$ represents how important the feature X_i is to the effort Y. In our work, $Rel(X_i,Y)$ is measured by normalized mutual information (NI) so it ranges from 0 to 1, as below:

$$Rel(X_i,Y) = NI(X_i,Y) \tag{5}$$

$Red(X_i, X_j)$ represents the redundancy between feature X_i and X_j. Similarly, it is defined as a normalized measure of feature redundancy: More redundant features have greater $Red(X_i, X_j)$ values, with minimum value 0 and maximum value 1. Recall redundancy is normally the correlation between features. In the proposed method, it is defined as the multivariate correlation between X_i, X_j and Y. In information theory, the conditional mutual information $CI(X_i; X_j \mid Y)$ measures the conditional redundancy provided Y. So the multivariate correlation can be explained by the difference between mutual information and conditional mutual information, which is $I(X_i; X_j) - CI(X_i; X_j \mid Y)$. Following above heuristic, $Red(X_i, X_j)$ can be defined the follow:

$$Red(X_i, X_j) = \frac{I(X_i; X_j) - CI(X_i; X_j \mid Y)}{I(X_i; X_j)} \tag{6}$$

The relevance weights $Rel(X_i, Y)$ and redundancy weights $Red(X_i, X_j)$ are utilized by the two sub-modules of R2D: For two projects p and q, $RelD(p,q)$ is the distance that accounts for feature relevance and $RedD(p,q)$ is the distance that accounts for feature redundancy. Following the heuristics in feature selection methods, redundancy should be subtracted from the distance, so R2D follows the form as below:

$$R2D(p,q) = RelD(p,q) - RedD(p,q) \tag{7}$$

Many distances can be utilized to define $RelD(p,q)$, such as weighted Euclidean distance and weighted Manhattan distance. In our work, $RelD(p,q)$ is defined as the square of weighted Euclidean distance:

$$\begin{cases} RelD(p,q) = \sum_{i=1}^{i \le m} w_i \delta_i^2 \\ \delta_i = \mid x_{p,i} - x_{q,i} \mid \end{cases} \tag{8}$$

Unfortunately, we don't have a widely accepted distance to define $RedD(p,q)$. In this paper, we provide some general rules to guide the design: (1) redundancy is the correlation between features and (2) the overall distance $R2D(p,q)$ should be non-negative. Following the rules, $RedD(p,q)$ is defined as Eq.9. Note m is the number of features and in case $m=1$, there's no redundancy between features and hence $RedD(p,q) = 0$.

$$RedD(p,q) = \sum_{i=1}^{i<m} \sum_{j>i}^{j \le m} \frac{2\delta_i \delta_j \sqrt{w_i w_j}\,\theta_{ij}}{m-1} \quad 0 \le \theta_{ij} \le 1, m > 1 \tag{9}$$

The constant factor $\frac{2}{m-1}$ ensures $R2D(p,q) \ge 0$ since:

$$\begin{aligned} & R2D(p,q) \times (m-1) \\ &= (m-1) \sum_{i=1}^{i \le m} w_i \delta_i^2 - \sum_{i=1}^{i<m} \sum_{j>i}^{j \le m} 2\delta_i \delta_j \sqrt{w_i w_j}\,\theta_{ij} \\ &= \sum_{i=1}^{i<m} \sum_{j>i}^{j \le m} (w_i \delta_i^2 + w_j \delta_j^2 - 2\sqrt{w_i w_j}\,\delta_i \delta_j \theta_{ij}) \\ &\ge \sum_{i=1}^{i<m} \sum_{j>i}^{j \le m} (\sqrt{w_i}\,\delta - \sqrt{w_j}\,\delta_j)^2 \ge 0 \end{aligned} \tag{10}$$

Denote $\Delta = \langle \delta_1, \ldots, \delta_m \rangle$ and $w_{ij} = \frac{-\sqrt{w_i w_j}\,\theta_{ij}}{m-1}$. Then R2D can be summarized as:

$$\begin{aligned} R2D(p,q) &= \sum_{i=1}^{i \le m} w_i \delta_i^2 + \sum_{i=1}^{i<m} \sum_{j>i}^{j \le m} 2\delta_i \delta_j w_{ij} \\ &= \Delta \begin{vmatrix} w_1 & w_{1,i} & w_{1,m} \\ & \ddots & \\ w_{m,1} & w_{m,i} & w_m \end{vmatrix} \Delta^T = \Delta W \Delta^T \end{aligned} \tag{11}$$

4. Experiment Setup

Experiments are designed to evaluate if *the proposed method R2D is an effective method*. For this purpose, R2D is compared against four conventional feature weighting techniques on six datasets drawn from the PROMISE repository [14].

4.1. Datasets

Six datasets with various characteristics are selected from the PROMISE repository [14] for experiments. Pre-processing steps, including data cleaning, normalization and feature selection (with mRMR [15]), are performed before feeding the datasets to estimation models. Table 1 summarizes the datasets.

Table 1: Dataset information

Dataset	Feature Number (Selected)	Project Number	Effort Range
Albrecht	7(6)	24	[0.5,105.2]
China	16(13)	499	[26,54620]
Desharnais	11(4)	77	[546,23940]
Kemerer	6(5)	15	[23.2,1107.31]
Kitchenham	5(4)	135	[219,113930]
Miyazaki	7(6)	48	[5.6,1586]

4.2. Comparative methods

While R2D is a filter method based on Euclidean distance, other available filter FWTs are selected to assign weights to Euclidean distance. Specifically, Mantel's correlation (Mantel) [12], correlation coefficient (Cor), normalized mutual information (NMI) and FWT with linear regression (LR) [16] are included for comparison.

All methods are used in combination with k Nearest Neighbor and an optimized k value is selected with 5-fold cross validation for each method. With the optimized k value, the Leave-One-Out Cross Validation (LOOCV) is applied for validation.

4.3. Evaluation metrics

Many metrics have been proposed in the literature, such as MAR, MMRE and Pred(25). However, MMRE has been criticised for being bias [17], hence we will only employ MAR and Pred(25) in this paper.

The Mean Absolute Residual (MAR), as defined in Eq.12, measures the absolute residual of the estimation. The smaller the MAR is, the more accurate the distance is.

$$MAR = \frac{1}{n}\sum_{i=1}^{n}|y_i - y_i| \tag{12}$$

Pred(25) is another widely used evaluation metric. It evaluates the percentage of estimates that are within 25% of the actual value so a higher value is preferred. Clearly, Pred(25) is insensitive to estimations that fall beyond the 25% accuracy. In fact, it measures the kurtosis which describes the extent to which the estimation peaks around its central value [17].

$$Pred(25) = \frac{1}{n}Count(MRE_i \le 25\%) \tag{13}$$

4.4. Statistical tools

We employ the Wilcox's percentile bootstrap method [18] to examine significant difference between R2D and comparative methods. Moreover, the Glass's Δ is provided to measure the magnitude of difference [19]. As defined in Eq.14, σ is the pooled standard deviation of absolute residue. Δ is considered to be small (≈ 0.2), medium (≈ 0.5) and large (≈ 0.8) [20].

$$\Delta = \frac{MAR_A - MAR_B}{\sigma} \tag{14}$$

5. Result Analysis

The estimation accuracy of R2D and comparative methods are listed in Table 2. Obviously, no method achieves optimal estimation accuracy on all datasets. Moreover, it's hard to draw a solid conclusion since we may get contradictory conclusion from various evaluation metrics.

To be specific, R2D achieves minimum MAR and optimal Pred(25) on the same 4 datasets (Albrecht, China, Kitchenham and Miyazaki) and minimum MMRE on 2 of the 4 datasets (China and Kitchenham). While R2D achieves optimal MAR and Pred(25) on the same datasets, the decrease of MAR values may be partly explained by Pred(25) since the improvement of Pred(25) will decrease the MAR value due to a higher chance of making 25%-accurate estimation. The improvement of MAR and Pred(25) against the second optimal method on the four datasets can reach 17.4% and 27.8% on average, respectively.

Table 2: Estimation accuracy comparison. Optimal values are bolded.

		Feature Weighting Techniques				
Measure	Dataset	Cor	LR	Mantel	R2D	NMI
MAR	Albrecht	7.27	9.02	7.12	**5.88**	7.28
	China	907	958	822	**586**	1082
	Desharnais	2448	2212	**2007**	2285	2082
	Kemerer	128	125	**110**	125	132
	Kitchenham	1835	1630	1947	**1369**	1688
	Miyazaki	55.9	55.3	55.9	**53.7**	55
Pred(25)	Albrecht	0.33	0.33	0.33	**0.46**	0.33
	China	0.58	0.56	0.63	**0.77**	0.52
	Desharnais	0.3	0.36	**0.45**	0.34	0.36
	Kemerer	0.4	0.33	**0.53**	0.47	0.47
	Kitchenham	0.31	0.41	0.29	**0.53**	0.44
	Miyazaki	0.29	0.31	0.29	**0.4**	0.29

To understand the statistical validity of the comparison, the median difference of absolute residual (AR) between R2D and comparative methods is presented in Table 3. For each pair-wise comparison, the probability of median difference $= 0$ is given by p. The upper and lower bounds give the 95% confidence interval for the median difference. Therefore, an interval that does not straddle zero indicates a significant difference.

Table 3: Median absolute residual (MAR) differences between R2D and comparative methods.

Dataset	Method	p-value	Median	Lower	Upper
Albrecht	Cor	56.90%	-87.00%	-327.00%	490.00%
	LR	0.7	-65.00%	-483.00%	3.25
	Mantel	61.80%	-57.00%	-528.00%	380%
	NMI	0.451	-0.27	-2.65	3.12
China	Cor	0	-86	-136	-35
	LR	0	-134.5	-224	-69
	Mantel	0	-68.5	-118.5	-38.5
	NMI	0	-125.5	-186	-63.5
Kitchenham	Cor	0.004	-197	-389	-34.5
	LR	0.068	-81	-337	16.5
	Mantel	0	-170	-492.5	-44.5
	NMI	0.32	-114	-416	87

As can be observed, there is significant difference for MAR on China and Kitchenham. Less significant difference can be identified on other datasets. We can observe from Table 1 that both Kitchenham and China has a large number of projects. As noticed by [19], significant difference can be identified provided sufficient data. Hence the statistical test may fail to identify significant difference due to insufficient data. As an example, Table 4 presents the MAR effect sizes of R2D against comparative methods. As can be observed, even though difference of MAR on Albrecht dataset is not significant by Table 3, the difference reaches small according to Glass's Δ (≈ 0.2). Hence the threats of statistical test to the conclusion should be addressed.

Table 4: MAR Effect size of R2D against comparative methods.

Albrecht	0.19	0.33	0.18	0.19
China	18.00%	20.00%	14.00%	25.00%
Desharnais	0.06	-	-	-
Kemerer	0.01	0.00%	-	3.00%
Kitchenham	0.05	0.03	0.07	0.04
Miyazaki	0.01	0.01	0.01	0.01

- stands for the case where comparative method outperforms R2D

From the above analysis, we can learn that redundancy weighting (R2D) is an effective method in a way that it achieves optimal MAR and Pred(25) on the same 4 out of 6 datasets.

6. Conclusion

This work is motivated by the assumption that the estimation accuracy may be further improved by addressing both feature relevance and redundancy in Case Based Reasoning (CBR). Following the assumption, a new distance measure called Relevance-Redundancy (R2D) is proposed to testify the assumption. Experiment results on six benchmark datasets indicate that the assumption is tenable to some extent.

Although the estimation accuracy can be improved by our assumption, current implementation is intuitive in essence. Hence we plan to apply some optimization techniques such as PSO to learn better relevance-redundancy weighting.

7. Acknowledgements

The paper is sponsored by The National Key Research and Development Program of China (2016YFB1000805).

8. References

[1] Barry Boehm, Chris Abts, and Sunita Chulani. Software development cost estimation approaches: a survey. *Annals of software engineering*, 10(1-4):177–205, 2000.

[2] Ali Idri, Fatima azzahra Amazal, and Alain Abran. Analogy-based software development effort estimation: A systematic mapping and review. *Information and Software Technology*, 58:206–230, 2015.

[3] Vahid Khatibi Bardsiri, Dayang Norhayati Abang Jawawi, Siti Zaiton Mohd Hashim, and Elham Khatibi. A flexible method to estimate the software development effort based on the classification of projects and localization of comparisons. *Empirical Software Engineering*, 19(4):857–884, 2014.

[4] Lefteris Angelis and Ioannis Stamelos. A simulation tool for efficient analogy based cost estimation. *Empirical software engineering*, 5(1):35–68, 2000.

[5] Jorge R Vergara and Pablo A Est ́evez. A review of feature selection methods based on mutual information. *Neural Computing and Applications*, 24(1):175–186, 2014.

[6] Isabelle Guyon and Andr ́e Elisseeff. An introduction to variable and feature selection. *The Journal of Machine*

Learning Research, 3:1157–1182, 2003.

[7] Lei Yu and Huan Liu. Efficient feature selection via analysis of relevance and redundancy. *The Journal of Machine Learning Research*, 5:1205–1224, 2004.

[8] Boyce Sigweni and Martin Shepperd. Feature weighting techniques for cbr in software effort estimation studies: a review and empirical evaluation. *In Proceedings of the 10th International Conference on Predictive Models in Software Engineering*, pages 32–41. ACM, 2014.

[9] Sigweni, Boyce B. An investigation of feature weighting algorithms and validation techniques using blind analysis for analogy-based estimation. Brunel University London. 2016

[10] Sun-Jen Huang and Nan-Hsing Chiu. Optimization of analogy weights by genetic algorithm for software effort estimation. *Information and software technology*, 48(11):1034–1045, 2006.

[11] Martin Auer, Adam Trendowicz, Bernhard Graser, Ernst Haunschmid, and Stefan Biffl. Optimal project feature weights in analogy-based cost estimation: Improvement and limitations. *Software Engineering, IEEE Transactions on*, 32(2):83–92, 2006.

[12] J. W. Keung and B. Kitchenham. Optimising project feature weights for analogy based software cost estimation using the mantel correlation. In *Software Engineering Conference*, 2007. APSEC 2007. 14th Asia-Pacific, pages 222–229.

[13] Pablo A Est ́evez, Michel Tesmer, Claudio A Perez, and Jacek M Zurada. Normalized mutual information feature selection. *Neural Networks, IEEE Transactions on*, 20(2):189–201, 2009.

[14] The promise repository of empirical software engineering data, 2015.

[15] Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(8):1226–1238, 2005.

[16] Emilia Mendes, Ian Watson, Chris Triggs, Nile Mosley, and Steve Counsell. A comparative study of cost estimation models for web hypermedia applications. *Empirical Software Engineering*, 8(2):163–196, 2003.

[17] Barbara A Kitchenham, Lesley M Pickard, Stephen G. MacDonell, and Martin J. Shepperd. What accuracy statistics really measure [software estimation]. In *Software, IEE Proceedings-*, volume 148, pages 81–85. IET, 2001.

[18] Rand R Wilcox. Introduction to robust estimation and hypothesis testing. *Academic Press*, 2012.

[19] Vigdis By Kampenes, Tore Dyb ̊a, Jo E Hannay, and Dag IK Sj øberg. A systematic review of effect size in software engineering experiments. *Information and Software Technology*, 49(11):1073–1086, 2007.

[20] Jacob Cohen. Statistical power analysis for the behavioral sciences. 2nd edn. hillsdale, new jersey: L, 1988.