JOURNAL OF AERONAUTICS AND SPACE TECHNOLOGIES
JULY 2015 VOLUME 8 NUMBER 2 (1-6)

PERFORMANCE EVALUATION OF PROJECTS IN SOFTWARE DEVELOPMENT

Filiz ÇETİN*

Turkcell Teknoloji, Teknoloji Plaza, Yeni Mahalle, Pamukkale Sokak, Soganlık, 34880 Kartal, İstanbul TURKEY, filiz.cetin@turkcell.com.tr

Çiğdem ALABAŞ-USLU

Endüstri Mühendisliği Bölümü, Marmara Universitesi, Göztepe kampus, 34722, İstanbul, TURKEY, cigdem.uslu@marmara.edu.tr

Received: *05th May 2015,* ***Accepted:*** *02nd July 2015*

ABSTRACT
IT firms are able to develop various types of software development projects from small sized projects to very large ones. A software development process is carried out by different stages of the project management such as analysis, design, development and testing. At the end of the process, performance of the project is evaluated by project sponsor who represents the customer of the project. There are different factors that effect the performance of the projects like risk, project size, project type and priority, team size, budget, duration, change requests and delays. In this study, we aim to statistically analyze effects of these factors on performance evaluation of the project sponsor. Additionally, we try to develop a statistical model to aid the project sponsor in performance evaluation. We use real data from software development department of telecommunication firm.

Keywords*: Software Development, Project Performance, Statistical Models.*

YAZILIM GELİŞTİRME PROJELERİNDE PERFORMANS DEĞERLENDİRMESİ

ÖZET
BT firmaları küçük boyutlulardan büyük boyutlulara kadar farklı yazılım geliştirme projelerini geliştirme konusunda çeşitli becerilere sahiptir. Bir yazılım geliştirme projesi analiz, geliştirme ve test gibi farklı proje yönetim katmanlarından oluşabilir. Projenin performansı aynı zamanda projenin müşterisi olarak da tabir edilen proje sponsoru tarafından sürecin sonunda belirlenir. Risk, proje boyutu, proje büyüklüğü ve önceliği, ekip büyüklüğü, bütçe, süre ve değişiklik istekleri gibi faktörler projenin performansını doğrudan etkileyebilir. Bu çalışmada, söz konusu faktörlerin proje sponsorunun performans değerlendirmesine olan etkileri istatistiksel olarak analiz edilmiştir. Bunun yanında makalede proje sponsorunun performans değerlendirmesini kolaylaştırmak amacıyla bir istatistiksel model de geliştirilmiştir. Çalışmada, bir telekomünikasyon firmasının yazılım geliştirme bölümünden alınan gerçek veriler kullanılmıştır.

Anahtar Kelimeler: *Yazılım Geliştirme, Proje Performansı, İstatistiksel Model.*

1. INTRODUCTION

Project is a temporary collection of efforts to create a unique product, service or result. Project has definite beginning and end, definite scope and aim within a limited budget [1]. Each project has a sponsor which is defined in [1] as “the person who is the entity giving resources and support to the project”. According to the methodology proposed in [1], the project is closed with the approval of the sponsor.

Software development projects are different and more difficult from other engineering projects [2], as their complexity and high rate of failure [3]-[4]. CHAOS manifesto by Standish group [5] represent that in 2012, 18% of software projects failed and 43% faced with challenges. A small portion of software projects has been found successful [5]. Therefore understanding and measuring software project success is a critical process in characterizing software development projects.

* Corresponding Author

Literature about project success is divided into four periods [6]-[7]: First is "Early 1970s"; success literature focuses on time, cost and quality. This dimension is defined as "iron triangle" [8]. Second is "1980s to 1990s" that looks at the technical aspects of a project to how it related to the client organization [9], and success was typically described with a single measure for the project instead of multiple measures different stakeholders such as customers, employees and managers assess project success in any organization. Project success is best judged by the stakeholders, especially the primary sponsor [13]. Prabhakar [14] notes schedule and budget performance are considered inadequate as measures of project success. Agarwal and Rathod [5] state that cost, time and quality are still important criteria for evaluating the performance of software project, however cost is considered to be the least important criterion for figuring a project's success. Procaccinoa & Verner [15] find that completing a project on time and completing it within budget, do not appear to have much relevance for the importance to the project success. According to the Muller & Judgev [7], there are differences in the rating of success criteria by project type. Muller & Judgev [7] explore the statistical behavior and associated relationships between project success factors, PM tools, and PM methods. In their research, Reyes et al. [3] aim toward over the life [7]. Third is "1990s to 2000s" that gives importance to critical success factors frameworks and internal and external stakeholders [10]. The "21st Century" is the last one and continuing today that researches focus on stakeholder with project success [11]-[13]. Davis [6] also emphases about this period that there is a high attention of owner and sponsor involvement in projects. Prabhakar [14] expresses that developing software project success/risk analysis models which can aid project managers in identifying, analyzing and controlling potential risks during software development. Because of the complexity of the project success concept discussed above, there is lack of consensus among authors in the field [16].

In this study, we look for the statistical effects of factors on performance evaluation of the project sponsor. Moreover, we develop a neural network model to be used to estimate project performance. In the first section we give factors which have influence on performance evaluation. Second section is about analysis of the factors statistically. Third section explains the tabu search training algorithm for neural networks. Fourth section is the implementation of the algorithm for the performance evaluation of the projects. Finally we give conclusions in the last section.

2. FACTORS IN PERFORMANCE EVALUATION OF THE PROJECTS

In this study, factors which may affect performance evaluation of projects are determined by examining the historical data and relying expert opinions in a software development company in Turkey. Throughout the study, "project" term is used to mean "software development projects". These factors and their short descriptions are listed in Table 1.

Table 1. Description of factors in evaluation of project performance.

Factor No.	Factor Name	Factor Description
F1	Type of project	There are many types of project such as infrastructure, service, product, feasibility. Every project must fall into one of these types.
F2	Methodology	Software development methodologies are described in here.
F3	Number of baselines	Baseline is a point of reference.
F4	Severity	Indicates the severity level of the project
F5	Duration of project	Indicates total elapsed time of the project
F6	Difference btw actual and planned baseline dates	Difference between the actual and planned dates
F7	Duration of analysis	Time span for analysis
F8	Duration of development	Time span for development
F9	Duration of test	Time span for test
F10	Total number of launches	Total number of launces
F11	Total number of issues	Total number of bugs and issues seen throughout the project
F12	Total number of risks	Total number of an events that may end up with a negative impact
F13	Total number of Change Request	A change request is a formal proposal for an alteration to some product or system
F14	Project team size	Size of the project team
F15	Occurrence of stand by	Number of occurrences of standbys (on-hold)

F16	Duration of stand by	Elapsed time until the project starts again
F17	Reason of stand by	The reason of the standby
F18	Complexity of project	The level of complexity of the project (can ben complex, moderately complex or not complex)

The historical data also contains a project sponsor grade for each project. Each project has a sponsor mostly a manager or executive having overall accountability for the project. The sponsor acts like a champion of the project, selling and marketing the project throughout the company. In this paper, it's assumed that performance of a project is evaluated by the project sponsor and the assessed grade by him/her is treated as a measure of the performance for the project.

3. STATISTICAL ANALYSIS OF THE FACTORS IN PERFORMANCE EVALUATION OF THE PROJECTS

In this section, a statistical analysis is carried out to show the factors which are statistically significant in the evaluation of project performances by the project sponsor. For this purpose, the main factors notated by F1 through F18 are analyzed by the linear regression at significance level of 5%. Result of the study indicates that the type of project (F1) and the duration of project (F5) are statistically significant in the the regression model of explanation of the grades assessed by the sponsor.

Once the main factors which statistically significant on the grades of projects are obtained, a further analysis is implemented to analyze the factors which effect these main factors. Regression analysis shows that the relationships between F1 and F4, F11, and F14 are statistically significant at level of 5%, whereas F2, F3, F6, F7, F8, F9, F11, F12, and F18 are statistically significant in a regression model of F5.

4. TABU SEARCH TRAINING ALGORITHM FOR NEURAL NETWORKS

Tabu search training (TST) algorithm is proposed by Dengiz et al. [17] as a new supervised-learning approach to train multi-layer perceptrons (MLPs) for estimation purposes. MLPs are a kind of neural networks which consists of layers. Each layer of an MLP contains different number of neurons and the neurons in successive layers are connected by weights (synapses). Dengiz et al. [18] also show the application of TST algorithm to obtain neural network metamodels for the optimization of two different manufacturing systems.

TST algorithm utilizes a short-term memory to prevent of cycling of moves and a longer-term memory for diversification purpose. The algorithm proceeds iteratively by repeating the neighborhood generation mechanism managed by the short and the long-term memories until a termination criterion is met. The neighborhood generation mechanism, utilization of the short-term and long-term memories, parameter optimization and other aspects of the algorithm can be found in Dengiz et al. [17] in detail. The distinguishing characteristics of TST algorithm are given below briefly. Pseduo-code of the algorithm is also presented in Figure 1.

- Vector of current weights, $W_{cur} = [w_1, w_2, ..., w_n]$, is initialized randomly from a uniform distribution in the range of [-0.5, 0.5].
- At each iteration only one weight, w_j, is increased or decreased by v_j drawn from a uniform distribution in the range of $[B_L, B_U]$ to provide a sensitive local search. Change in a weight is called a move and resulting weight vector is a neighbor. Neighborhood of W_{cur} contains nK neighbors obtained by changing each weight n K times.
- The algorithm iterates by moving from current solution to the best available neighbor with the minimum RMSE (root mean squared error) given in equation 1, if the move which creates this neighbor is allowed through the tabu mechanisms.
- Short-term memory consists of tabu lists (mechanisms). Both the indices of modified weights and the amounts of change in weights are kept in related tabu lists to avoid from cycling around local optima. Tabu list, *tvs*, records v_j values for each weight, w_i, through the last V iterations. The list *tvs* forces the generation of random v_j values to be sufficiently different from the recorded V values. If v_j is near to one of V values then B_L and B_U bounds are increased by Δ at a time. Tabu lists, *tis* and *tds*, keep track of weight indices to prohibite increasing of a weight which decreased during last S iterations and vice verca.
- Frequency information about the highly repeated moves is recorded in a long-term memory, *freq*, to extend the search to the unexplored regions of the solution space. According to the long-term memory a weight w_j changed F times during L successive iterations is prohibited to give a chance the moves which made infrequently.

5. IMPLEMENTATION OF TABU SEARCH TRAINING ALGORITHM FOR PERFORMANCE EVALUATION OF THE PROJECTS

In this section, TST algorithm is used to develop an NN model to estimate performance evaluations of the projects. Structure of the NN model is MLP in which one input, one hidden and one output layers exist. Totally 13 factors which have been found as important on performance of evaluations of the project sponsors are the inputs of the MLP. Therefore 13 input neurons occur in the input layer to each of the inputs. Output layer consists of only one neuron and output calculated from this neuron is the estimation of performance evaluation with respect to the introduced inputs to the model. Additionally, one bias factor also must be included to both input and hidden layers. Structure of the MLP is given in Figure 2. Only one remaining question about the structure is the number of neurons in the hidden layer. To find the number of neurons in hidden layer experimentally, MLP model is trained separately for varying number of hidden neurons. Available data to develop the MLP contains 855 exemplars. Each exemplar is associated with the 13 factors and one grade which represents performance evaluation of the project sponsor. To select the best size of hidden layer, 570 exemplars out of 855 is used for training purpose and the remaining 285 for testing purpose. The algorithm aims to minimize RMSE over the training set. The error, which is used to calculate RMSE, is the difference between the grades by project sponsor from the historical data and the estimation of the grade by the MLP. Results from this fine-tuning study is summarized in Table 2. Table 2 shows that training and testing RMSE values according to varying number of the hidden neurons are close to each other. This results indicates robustness of TST algoritm against the hidden layer size. Nevertheless, the hidden size is set to 12 neurons since the lowest testing RMSE is obtained by using 12 hidden neurons.

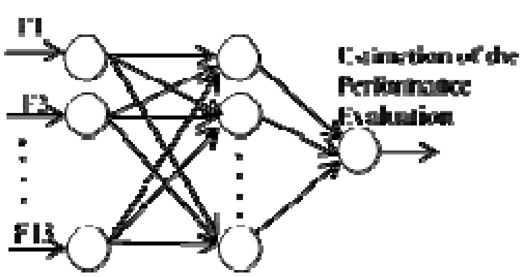

Figure 2. Structure of MLP to estimate the performance evaluation.

Table 2. Training and testing RMSE values according to varying number of the hidden neurons.

# of hidden neurons	Training RMSE	Testing RMSE	Average of training and testing RMSE
6	0.00351	0.00429	0.00390
7	0.00354	0.00427	0.00390
8	0.00351	0.00425	0.00388
9	0.00353	0.00425	0.00389
10	0.00352	0.00431	0.00392
11	0.00350	0.00429	0.00390
12	0.00352	0.00424	0.00388
13	0.00350	0.00428	0.00389
14	0.00351	0.00426	0.00388

Once the structure of the MLP is constructed, TST algorithm is employed for training purpose using 570 patterns that are randomly picked from the historical data of the software development company and then the MLP is tested on an additional 285 patterns. A four-fold cross validation technique is used to validate the MLP. Cross validation is a standard tool in statistics to validate statistical predictions [19]. Table 3 represents the results of the four-fold cross validation in terms of the training and testing RMSE values achieved by TST algorithm. TST algorithm is run for 5 replications and 5000 iterations per replication. As standard deviation of the RMSE values of the five replications is quite small, the number of replications is found enough. Average RMSE values given in Table 3 shows the validity of the MLP to estimate performance evaluations of the projects.

Table 3. Cross-validation results.

Folds	Training RMSE	Testing RMSE
Fold 1	0.00340	0.00445
Fold 2	0.00353	0.00422
Fold 3	0.00335	0.00434
Fold 4	0.00351	0.00411

To represent extrapolation capability of the MLP model, the model is also tested on an additional data (including 350 patterns) which are not used in both training and testing sets. TST algorithm is utilized to train the MLP over 855 patterns. The best MLP structure from the five replications of TST algorithm is then tested on the additional data. Result of this experiments shows that the training RMSE is 0.00356 while the extrapolation RMSE is 0.00445.

$$RMSE = \sqrt{\frac{\sum_{e=1}^{E}\sum_{j=1}^{J}\left(d_{ej} - o_{ej}\right)^2}{EJ}} \quad (1)$$

Where d_{ej} and o_{ej} are the actual output and estimated output of the j^{th} node in the output layer, respectively. E is the number of exemplars in the training set and J is the number of nodes in the output layer.

Initialize TST parameters

Repeat

For i = 1 *to n*

For j = 1 *to K*

$B_U \leftarrow \bar{B}_U$

$B_L \leftarrow \bar{B}_L$

Repeat

Generate $v_{i,j} \leftarrow$ Uniform $[B_L, B_U]$ or $v_{i,j} \leftarrow$ - Uniform $[B_L, B_U]$

Generate the neighbor $W^{((i-1)K+j)}$ and calculate the RMSE value

If $\left|v_{i,j} - tvs_i\right| \le \left(\frac{B_U - B_L}{2}\right)$ for any element in tvs_i *then*

$B_U \leftarrow B_U + \Delta$, $B_L \leftarrow B_L + \Delta$, tabu_status ← true

If ($v_{i,j} > 0$) and ($iter \le tis_i + S$) *then* tabu_status ← true;

If ($v_{i,j} < 0$) and ($iter \le tds_i + S$) *then* tabu_status ← true;

If ($freq_j > F$) *then* tabu_status ← true;

End If

Until (tabu_status ← false) or (RMSE{$W^{((i-1)K+j)}$} < RMSE{W^*})

Next j

Next i

If $W^{((i-1)K+j)}$ is the best neighbor with minimum RMSE *then*

update W_{cur}, tabu lists tds_i or tds_i and tvs_i

r ← r +1 if r < L then update $freq_j$ else r ← 0 and void $freq_j$

End If

Until a termination condition is met

Figure 3. Pseduo-code of TST algorithm.

6. CONCLUSIONS

In this study, an NN model, MLP, for performance evaluations of the projects in a software company is developed. Although there exist numerous statistical approaches proposed to evaluate performances of projects in the related literature, NNs are used rarely. A tabu search based supervised learning approach by Dengiz et al. [18] is used to train the MLP for estimation of project performances. Data used to train and test the MLP is derived from the historical data of the company. Results obtained from the cross-validation show that proposed MLP model is valid for the company. The MLP trained using the complete data also is tested on an additional data which are not used in both train and test set to show the generalization capability of the proposed approach. The experimental study indicates that proposed MLP model can be utilized to estimate the project performances and therefore can aid the decision maker to self-evaluate his/her decisions on the projects.

7. REFERENCES

[1] Pmbok, 2013, A Guide to the Project Management Body of Knowledge (PMBOK® Guide)—Fifth Edition.

[2] Repiso, LR, Setchi, R., Salmeron, JL, 2007, Modeling IT projects success: Emerging methodologies reviewed. *Technovation*. 27, 10, 582-594.

[3] Reyes F, Cerpa N, Candia-Vejar A, Bardeen M, 2011, The optimization of success probability for software projects using genetic algorithms, *Journal of System and Software,* 84(5): 775-785.

[4] Standish Group, 2013 Choas manifesto 2013: Think Big, act Small.

[5] Agarwal, N., Rathod, U., 2006. Defining "success" for software projects: an exploratory revelation. *International Journal of Project Management*, 24, 358–370.

[6] Davis, K., 2014, Different stakeholder groups and their perceptions of project success, *International Journal of Project Management,* 32,189-201.

[7] Müller, R., Jugdev,K. ,2012, Critical success factors in projects, *International Journal of Managing Projects in Business*, Vol. 5 Iss 4 pp. 757 – 77.

[8] Atkinson, R., 1999, Project management: cost, time and quality, two best guesses and a phenomenon, its time to accept other success criteria, *International Journal of Project Management*, 17, 337--342.

[9] Pinto, J.K., Slevin, D.P., 1988, Project success: definitions and measurement techniques. *Project Management Journal*, 19 (1), 67–73.

[10] Lester, D.H., 1998, Critical success factors for new product development, *Research Technology Management* 41 (1), 36–43.

[11] Turner, J.R., 2004, Five conditions for project success*, International Journal of Project Managemen*t 22 (5), 349–350.

[12] Turner, J.R., Zolin, R., Remington, K., 2009, “Modelling success on complex projects: multiple perspectives over multiple time frames”, in: Gemuenden, H.-G. (Ed.), *The Proceedings of IRNOP9, the 9th Conference of The International Research Network of Organizing by Projects*, Berlin, June. Technical University of Berlin, Berlin.

[13] Turner, J.R., Zolin, R., 2012, Forecasting success on large projects: developing reliable scales to predict multiple perspectives by multiple stakeholders over multiple time frames, *Project Management Journal* 43 (5), 87–99.

[14] Prabhakar, G.P., 2008, What is project success: a literature review, *International Journal of Business and Management*, Vol. 3 No. 9, pp. 3-10.

[15] Procaccinoa J.D., Verner J. M., 2006, Software project managers and project success: An exploratory study, *The Journal of Systems and Software*, 79,1541–1551.

[16] Berssaneti F. T. & Carvalho M. M., 2014, Identification of variables that impact project success in Brazilian companies*, International Journal of Project Managemen*t.

[17] Dengiz, B., C. Alabas-Uslu and O. Dengiz, 2009, Optimization of manufacturing systems using a neural network metamodel with a new training approach, *Journal of the Operational Research Society*, 60(9), 1191-1197, 2009.

[18] Dengiz, B., C. Alabas-Uslu. and O. Dengiz, 2009, A tabu search algorithm for the training of neural networks, *Journal of the Operational Research Society*, 60(2), 282-291, 2009.

[19] Stone, M., 1974, Cross-validatory choice and assessment of statistical predictions, *Journal of the Royal Statistical Society Ser. B*, 36, 111–147.

VITAE

Filiz ÇETİN

Studying in Marmara University Engineering Management as a doctorate student, Çetin has previously completed Middle East Technical University Information Systems department with a masters degree. Currently Çetin works in a leading telecom operator in Turkey as project and program manager.

Çiğdem ALABAŞ-USLU

Alabaş-Uslu is an associate professor at Marmara University. She received doctorate degree in Industrial Engineering at Gazi University. Her studies mainly includes optimization techniques and heuristic algorithms.