# Toward Objective and Quantitative Assessment and Prediction of Teamwork Effectiveness in Software Engineering Courses

Shihong Huang
Computer & Electrical Engineering and Computer Science
Florida Atlantic University
Boca Raton FL 33431, USA
shihong@fau.edu

Dragutin Petkovic,
Kazunori Okada,
Marc Sosnick, Shenhaochen Zhu
Dept. of Computer Science
San Francisco State University
1600 Holloway Ave, TH 906
San Francisco CA 94132 USA
Petkovic@sfsu.edu

Rainer Todtenhoefer
Univ. of Applied Science, Fulda
Marquardstr. 35
D - 36039 Fulda, Germany
rainer.todtenhoefer@informatik.hs-fulda.de

## Forward

Today's software engineering projects require teamwork which students practice in upper division software engineering courses. However, do they really 'learn' teamwork practices? This month's column addresses this question. While reading this article please think about how the concepts presented might be effectively applied in a corporate setting. *Mark & Peter*

## Introduction

One of the critical challenges in effective software engineering (SE) education is the lack of objective assessment methods of how well student teams learn the critically needed teamwork practices, defined as the ability: (i) to learn and effectively apply *SE processes* in a teamwork setting, and (ii) to work as a team to develop satisfactory software (SW) *products*. In addition, there are no effective methods for *predicting* learning effectiveness in order to enable early intervention in the classroom. This is further complicated with the emergence of global SW teams. Current approaches to assess achievement of SE teamwork skills rely on qualitative and subjective data taken as surveys at the end of the class with only rudimentary data analysis. In this article we present initial progress in our research to address the *assessment and prediction* of student learning of teamwork effectiveness in SE education. Our novel approach is based on: a) extracting only *objective and quantitative* student team activity data during their team class project; b) pairing this data with related independent observations and grading of student team effectiveness in SE process and SE product components in order to create "training database"; and c) applying a machine learning (ML) approach, namely random forest classification (RF), to the above training database in order to create ML models, ranked factors and rules that can both explain and assess, as well as provide *prediction* of the student teamwork effectiveness. Student team activity data are collected from ongoing and synchronously offered SE classes at San Francisco State University (SFSU), Florida Atlantic University (FAU) and Fulda University, Germany (Fulda), for approximately 80 students each year, working in about 15 teams, where student teams are both local and global (with students from multiple schools). In this article we summarize our approach and present preliminary data analysis results which served to test the concept, data gathering and ML tools we intend to use. We believe that success in this project will transform teaching (e.g. assessment) of critically important SE teamwork and will be of benefit to managing SE projects in industry.

## The need for teamwork quantitative assessment

Modern SE practice involves the development of software in teams (often globally distributed) with the goals of developing software (SW) on schedule and budget, satisfying specifications, which is at the same time maintainable and delights the customers. The need for improved teaching and training in software development teamwork skills is evidenced by statistics on the unacceptably high incidence of failure of industrial software projects: about 9% are abandoned, about 1/3 fail, and over half experience cost and schedule overruns [1-5]. The evidence also indicates that these failures stem primarily from failures in communication, organization and teamwork aspects of SE and are not due to the software technology [1,4-8]. The ever-increasing globalization of software development puts even heavier strains on team communications. Understanding how to effectively teach and assess the achievement of teamwork skills in SE projects is thus critical.

Although teamwork is currently an integral component of many SE courses, questions of how to effectively assess the efficacy of students' learning of teamwork, and how to predict (and correct) student teams' failures remain. Most of the current literature on student learning and assessment of SE teamwork skills relies solely on qualitative and subjective data from class surveys and instructor observations at the end of the academic term [18,19]. While valuable, these assessment instruments and methods are difficult to use consistently and repetitively, and the accompanying data analysis methods fail to address complex interactions among team members and the tools they are using (e.g. for communication, code management, issue tracking). The absence of objective, quantitative and comprehensive data on student team activities (e.g. team communication dynamics; statistics of the usage of software development tools) leaves team communication patterns understudied and poorly understood. The fact that the assessments are performed only at the end of the class term also precludes early instructor intervention. Sophisticated automated machine learning (ML) techniques that are now regularly applied in bioinformatics, medicine, Web data mining, marketing, analysis of customer behavior, and even in SE for software quality assessments [9-11] have not been applied to this problem.

In our project we strive to address the assessment and prediction of student learning of teamwork effectiveness in SE education. First, we define SE teamwork in two components, as the ability: (i) to learn and effectively apply *SE processes* in a teamwork setting, and (ii) to work as a team to develop satisfactory *software (SW) products*. Our approach is novel in that it is based on: a) extracting only objective and quantitative student team activity data during their team class project; b) pairing these data with related independent observations and grading of student team effectiveness in SE process and SE product components in order to create a "training database"; and c) applying ML, namely random forest classification (RF) [17], to this training database to create predictive classification models, ranked factors and rules that can both explain and assess, as well as provide prediction of the student teamwork effectiveness. Since 2006, student team activity data have been collected in both local and global collaborative SE classes at San Francisco State University (SFSU), Florida Atlantic University (FAU) and Fulda University in Germany, from approximately 80 -100 students each year, working in about 20 teams of 5 students each [12-16]. Comprehensive research and data collection, as described here, were started in August 2012 [22]. Below we present details of our approach, followed by the status, some challenges and initial progress.

## An objective, quantitative assessment of teamwork effectiveness in SE education using ML

Our approach has three distinct steps:

*Step 1: Collection of the data on student team activity*

Quantitative data (measures) pertinent to student teamwork activity are collected during our joint SE classes while students are actively engaged in intensive team projects. This joint SE class is being taught in a synchronous manner at SFSU, FAU and Fulda. All teams of 5-6 students develop the same project and fulfill the same five synchronized project milestones using the same modern SE tools (e.g., e-mail server, Bugzilla, SVN), all running on Amazon Cloud to ensure easy access and maintenance. Student teams are formed based on a student skill self-survey such that the level of combined expertise and gender mix are approximately equal across teams, to factor out the students' prior skill level from this study. Instructors maintain a log of their regular observations about the teams which are later used for assessment and grading. All data we collect about student teamwork activities are: i) quantitative and objective; ii) related to measurable manifestations of teamwork activity; iii) easy to collect; and iv) amenable to analysis by machine learning methods. Student Activity Measures (SAM) focus on the activity of each student and are obtained by weekly online surveys and analysis of usage of SE tools. These are only quantitative measures, such as time used for certain project activities (e.g. coding, meeting, documentation), counts of e-mail, incidents, time to close on an open issue etc., and are measured by automated tools or easily observed by instructors or input by students via weekly surveys. Team Activity Measures (TAM) are obtained for each team by aggregating the SAM for the team's members. For example, a SAM datum is the number of commits to the team's source code repository; the corresponding TAM is the average and standard deviation of commits for all the team members. To examine different patterns of behavior at different stages of project development a time variable related to each of five project milestones is introduced. We believe that by focusing only on quantitative variables and combining them at the team level we reduce the influence in reporting errors and significantly eliminate subjective bias. All student personal information is removed from the databases to ensure privacy.

*Step 2: Creation of ML training database*

At the end of the semester, instructors and independent evaluators (faculty who do not teach the SE classes) evaluate/grade each student team for achievement of SE teamwork outcomes using the class grading rubrics which include student adherence to SE process as well as quality of developed SW product. These grades, one for adherence to the SE process and one for the quality of the team's SE product, are categorized for the purpose of our research and for easier ML implementation only in three categories or classes: "A - above expectations", "C – at expectations", or "F - below expectations". These grades constitute "decision classes" for the ML algorithm, and are paired with TAM data for each team to construct a ML training database.

*Step 3: Applying ML to discover factors that determine and predict student SE teamwork achievement.*

We use this training database as an input to ML training, in order to produce a ML classifier that predicts the student team performance based on TAM data, and can assess the effectiveness of TAM measures by evaluating ranked TAM factors. For ML technology we chose the random forest (RF) [17] for its accuracy, success in many application areas, and ability to generate simple rules that explain its behavior. We use open source SW for statistical computing, R [21] which offers an easy-to-use RF implementation as in [20]. We envision that our results will include: classification models to predict team performance; ranked factors and rules that are the best predictors of student team success; recommendations (e.g. polices, SW) for streamlining or automating the assessment and prediction of teamwork learning; and finally the training database of our measurements to enable others to apply and experiment with their own data ML analysis and methods.

**Status and Challenges**: We are in the process of collecting complete SAM and TAM data from a currently ongoing SE class offered jointly at SFSU, FAU and Fulda University. There are 20 student teams of 5-6 students each, out of which four are global teams: two global teams are comprised of SFSU and FAU students, and two global teams are comprised of SFSU and Fulda students. This year student teams are developing a website application for sharing cooking recipes targeted to student populations. All the data collection tools and ML techniques are implemented and data collection from our joint SE class is ongoing. The RF tools and SW are operational and integrated with data gathering. In order to evaluate and prepare/adapt the data gathering and RF tools, we conducted a pilot experiment with a preliminary TAM dataset collected from our SE classes earlier in 2010. This dataset consists of 12 teams and includes 9 factors that are a subset of the TAM design described in this article. The RF classifier is then trained to predict each team's SE process grade. Without any systematic parameter tuning, the number of feature subsets (mtry) and the number of trees (ntree) for RF are set to 6 and 50, respectively. We record an approximate generalization error [17] of 18.2% or equivalently 81.8% accuracy, demonstrating a preliminary proof-of-concept for our ML approach. More systematic data analyses are expected in 2013 when we collect the first batch of full TAM data from our ongoing class. In the course of this research we also evaluated the appropriateness of the data we are collecting and we verified that all are feasible to collect. In addition we constantly evaluate possible new measures we could use. To this effect, upon formally evaluating students teams in their first implementation milestone we noticed that student teams that were not making sufficient progress also provided poor (e.g. repetitive) or no code submit and no code header documentation. Therefore, we added to our list of TAM data a new measure tracking this issue (again, preserving the requirement that each measure is quantitative and objective). The new item is percent of code submits that have non-empty and unique comments, measured for the team at the end of each implementation milestone.

We still face some challenges:

- Sample size set (e.g. size of training database), where each sample refers to a student team, is very small. Given that from each team we extract about 20 TAM measures, there is a danger of not being able to reliably train RF algorithms. With time we plan to collect more data and keep refining our ML approach to adapt it to this small sample case.
- As part of our teaching commitment we help student teams by coaching them, and this process could bias and influence student team behavior and consequently the training data. We attempt to mitigate this by carefully taking records of our interventions and student actions and incorporating this in data analysis, as well as extracting data in certain predetermined time intervals. This allows us to factor in the impact of each measure before and after the intervention.

**Progress**: In our preliminary work we have implemented and tested all tools for data collection and ML processing and obtained some "proof-of-concept" results from the data from prior classes. The work on the project is ongoing with first results expected in 2013 once sufficient data from ongoing SE classes is collected and analyzed.

**Impact**: The proposed project has the potential to transform the teaching of practical SE skills by addressing SE teamwork, a major factor in the unacceptably high rate of failure of industry SW projects and a critical issue for maintaining the competitiveness of the US SW industry in the era of global SW development. Its prediction capability promises to help teaching SE by allowing early intervention for teams which underperform.

This research has been funded by NSF TUES Grant # 1140172

## REFERENCES


[1] "Standish Group Report: CHAOS Summary 2009." Internet:
[2] http://www1.standishgroup.com/newsroom/chaos_2009.php [May 18, 2011].
[3] C. Sauer, C. Cuthbertson, "The State of IT Project Management in the UK 2002-2003." Computer Weekly, 15 April 2003. London.
[4] C. Sauer, A. Gemino, B. Reich," The impact of size and volatility on IT project performance," Communications of the ACM 50(11). November, 2007, pp. 79-84.
[5] C. Jones, Software Engineering best Practices: Lessons from Successful Projects in the Top Companies. McGraw Hill, 2010, ISBN 978-0-07-162161-8.
[6] J. Reel, "Critical success factors in software projects," IEEE Software 16(3). 1999. pp. 18-23.
[7] R.N. Charette, "Why software fails," IEEE Spectrum 42(9). September 2005, p. 42.
[8] R. Pressman, Software Engineering: A Practitioner's Approach, Sixth Edition, McGraw Hill, 2005.
[9] B. Curtis, H. Krasner, N. Iscoe, "A field study of the software design process for large systems." Communications of the ACM 31(11), 1988, pp 1268–1287.
[10] P. Baldi, S. Brunak, Bioinformatics: The Machine Learning Approach, Second Edition (Adaptive Computation and Machine Learning), MIT Press, 2001
[11] P. Perner, ed., "Advances in data mining: applications in medicine, web mining, marketing, image and signal mining," 6th Industrial Conference on Data Mining, ICDM 2006. Lecture notes in Computer Science, Vol. 4065. Springer. 2006.
[12] E. Zhang, J. Tsai, ed., Machine Learning Applications In Software Engineering. Series on Software Engineering and Knowledge Engineering, Vol. 16. World Scientific, 2005.
[13] E. Petkovic, R. Todtenhöfer, G. Thompson, "Teaching practical software engineering and global software engineering: case study and recommendations," Proceedings of the 36th ASEE/IEEE Frontiers in Education Conference, San Diego, CA, 2006, pp. 19–24.
[14] D. Petkovic, G. Thompson, R. Todtenhöfer, "Teaching practical software engineering and global software engineering: evaluation and comparison." Proceedings of the Eleventh Annual Conference on Innovation and Technology in Computer Science Education, Bologna, Italy, 2006, pp. 294–298.
[15] D. Petkovic, G. Thompson, R. Todtenhöfer, "Assessment and comparison of local and global SW engineering practices in a classroom setting," Proceedings of the 13th Annual Conference on Innovation and Technology in Computer Science Education, Madrid, Spain, June 2008, pp. 78–82.
[16] G. Thompson, D. Petkovic, S. Huang, R. Todtenhöfer, "Teaching distributed collaborative development techniques in a software engineering class setting." Integrating FOSS into the Undergraduate Computing Curriculum (FOSS 2009). Chattanooga, 2009.
[17] D. Petkovic, G. Thompson, R. Todtenhöfer, S. Huang, B. Levine, S. Parab, G. Singh, R. Soni, S. Shrestha : "Work in progress: e-TAT : online tool for teamwork and "soft skills" assessment in software ngineering education," Frontiers in Education (FIE) 2010, IEEE. 27-30 Oct. 2010, pp. S1G-1-S1G-3,
[18] L. Breiman, "Random Forests," Machine Learning 45(1). 2001. pp. 5–32.
[19] "CATME - Comprehensive Assessment for Team-MemberEffectiveness." Intenet: http://www.catme.org [June 6, 2012].
[20] D. Davis, M. Trevisan, P. Daniels, K. Gentili, C. Atman, R. Adams, D. McLean, S. Beyerlein, " A Model for Transferable Integrated Design Engineering Education", World Federation of Engineering Organizations , 2003
[21] L. Breiman, A. Cutler , "Random Forest: Breiman and Cutler's random forests for classification and regression." Internet: http://cran.rproject.org/web/packages/randomForest/ [June 24, 2012].
[22] "The R Project for Statistical Computing." Intenet: http://www.rproject.org/ [June 24, 2012].
[23] D. Petkovic, K. Okada, M. Sosnick, A. Iyer, S. Zhu, R. Todtehoefer, S. Huang: "A Machine Learning Approach for Assessment and Prediction of Teamwork Effectiveness in Software Engineering Education", Frontiers of Education FIE 2012, Seattle, WA, October 2012