JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 33, 1447-1463 (2017)
DOI: 10.6688/JISE.2017.33.6.4

Making Programmer Effective for Software Development Teams: An Extended Study

ABDUL REHMAN GILAL[1], JAFREEZAL JAAFAR[2], AHSANULLAH ABRO[3],
WAHEED ALI UMRANI[4], SHUIB BASRI[5] AND MAZNI OMAR[6]
[1,2,3,5]*Department of Computer and Information Sciences*
Universiti Teknologi Petronas
Perak, 32610 Malaysia
[1,3,4]*Department of Computer Science*
Sukkur IBA University
Sukkur, 65200 Pakistan
[6]*School of Computing*
Universiti Utara
Kedah, 06010 Malaysia
E-mail: {a-rehman[1]; ahsanullah.abro[3]; waheed.ali[4]}@iba-suk.edu.pk;
{jafreez[2]; shuib_basri[5]}@utp.edu.my; mazni@uum.edu.my[6]

The fast growing demand of software has caused numerous challenges for software developers to produces quality software within deadlines. The main purpose of this research article was to find the suitable personality type combinations of programmer with team-leaders and programmers by gender classification in software development teams. Myers-Briggs Type Indicator (MBTI) was applied to measure the personality types of the study participants. In order to find the possible combination of personality types between team-leader and programmer, this study applied Genetic Algorithm (GA) and Johnson's Algorithm (JA) on data. Results emanated from training experiments were validated with Standard Voting (SV), Voting with Object tracking, and Naïve Bayes classification techniques based on prediction accuracy. Basically, two types of decision rules were formed: rules without gender classification of programmer but they only discussed the personality types of team-leader and programmer. Whereas, the second type of rules were composed of team-leader, programmer personality types, and gender of programmer. It was found that extrovert (E) trait programmers can be suitable with E trait team-leaders. In the same way, male programmer can work in a good way with male leaders or other way around for females. At the end, there were only certain personality types appeared to be effective in mixed gender teams.

Keywords: personality types, MBTI, team leader, programmer, gender, software development, team composition, rule-based

1. INTRODUCTION

The successful and progressing organizations of the world have attributed quality software as one of the key sources to progress by leaps and bound. By now, there is fueling strong demand of software for different purposes. But, it is regretfully added that the supply of quality software is decreasing drastically from software development workplaces. For instance, the past studies by Wysocki [1], Standish-Group [2], Glass [3], and Keil *et al.* [4] have shown their concern over the ratio of IT development projects that have continuously failed to achieve their desired ends. In the same vein, the Standish

Received December 29, 2016; revised March 6, 2017; accepted May 22, 2017.
Communicated by Kashif Saleem.

Groups [5] carried out longitudinal research study to determine the fall of successful software development in which several software development workplaces were surveyed. The findings of this study revealed that just 6% software serve the organizational purposes whereas 42% software become ineffective during developing process and majority of 52% software are challenged for their irrelevancy, time consumption, and expensive nature.

There could be several other reasons causing software development failure but the human aspect is considered the most substantial reasons of all. It is believed [6] that the development of software does not demand a developer with technical skills only because it is a socio-technical job which involves both technical and social expertise to develop the software. Hence, the software development can be successful if it integrates team members with technical and social skills. Similarly, Capretz & Ahmed [7] also asserted that the software development team becomes an ideal if its members are equipped with technical and social skills. Additionally, isolation of skills either social or technical will never ensure the quality development for software [8].

Keeping in view the grave importance of team composition for developing successful software, the current study suggests the rules for effective software team composition in order to reduce ambiguities. For this purpose, MBTI personality type is used to assess personality types amongst software team members. The MBTI personality types is chosen because it is widely used and accepted amongst researchers in the field of software engineering. In addition, researches regarding personality are giving lack of attention on gender perspectives, especially in software development, whereby it is quite hard to find theories, valid information and meaningful studies about gender [9]. This is due to the barrier of limitations on personality research occurred since previous days till now. During the studies of personalities in the first several decades, less attention was given on the important aspects and issues that related to women. Other important issues such as sexual orientation, ethnicity, and individual differences are being neglected as well. Usually, the studies were more focused on the white men as the samples whereby the hypothesis gained were bias on the white, androcentric point of view. However, researchers concluded that the data collected are valid and can be applied generally to all human personality. Researchers also paid less attention on the gender differences when women were included in the studies conducted and they were often unrecorded [9]. In order to cover the gender side of personality, this study has included the gender as an independent variable.

Additionally, the results emanating from this research study only benefits the teams comprising of four to six members who develop a small-scale software project. This study only focuses on the role of "programmer" with "team-leader" and "other programmers in the teams" which are the most important contributors in software development teams. Moreover, this paper is an extended piece of our previously published work [10]. The published work focused to provide the personality equations of programmer role with team-lead role only. Whereas, this extension includes the discussion on the personality types of programmer with other programmers. The following subsection of the paper are organized to highlight the related work, methodology, results and discussion, threats to validly, conclusion and future work.

2. RELATED WORK

No one can claim that the success of software could only be achieved by set of technical principles. In fact, yet there is no specific way of constructing software since each

and every application has its explicit demands. It is, therefore, not an easy task to identify and encounter all possible ambiguities which can occur during the process of software development.

The past research studies on software development also suggest that software development is a complex sociotechnical activity [11-13]. The vast research stream mainly asserts that software development is not only to develop the technical product but it also includes the social process which involves several actors with different backgrounds working together to achieve the same goal [14-16]. In the same vein, Curtis *et al.*, [17] maintained that the software development process should be treated as communication, negotiation, and learning activity. Therefore, it is the main objective of this study to see the effective equations of team members' composition with personality types which can ensure the learning, communication, and negotiation processes effectively. Moreover, as software development involves people in process, therefore issues related with human can never be ignored in this regard.

Numbers of research studies have been conducted in the past for finding the relationship between social and technical skills. In them, software development team composition with personality type of the members is also very impacting area. But unfortunately, unlike social science, the research on software development team composition has not reached at the maturity level [18, 19]. Moreover, past studies have left and created several ambiguities for practitioners to use the suitable team composition models for building ideal teams. For instance, Gorla and Lam [20] have recommended the E (extrovert) personality trait for programmer whereas Ahmed and Capretz [21] suggested the I (introvert) personality trait. For the same issue, Cruz and da Silva [22] and MacDonell [23] claim that the validity of models for effective team composition of software development is still dubious.

Moreover, to cope this prevailing problem in software team composition, Gilal *et al.*, [19] found the relationship between personality types together with gender for software development teams' composition. This study investigated the performance variations among software development team members caused by genders' personality types. For instance, the male-dominated teams created the reasons for the females for being ineffective in teams if the personality type of female was with "E" trait. Furthermore, the study revealed that the female-leader were more convenient with only female or majority-female (*i.e.*, having female in majority) groups. On the contrary, male-leaders were adjustable with all kind of team compositions. Critically, this study was just based on tabulated calculation and could not give any statistical or predictive evidences. However, this study also opened new vista for further research on the gender with personality types to reach appropriate conclusions. Moreover, the study by Mazni *et al.*, [24] also studied the relationship with project success and personality types. This study focused the homogeneous and heterogeneous personality based teams' performance and its findings suggested that heterogeneous teams were more effective than the homogeneous teams.

In order to measure the personality type of an individual the MBTI has been widely used in social sciences. It is also reliable and valid tool among software engineering researchers [19, 25-28]. For instance, Furnham [29] used MBTI in training and consultancy areas, and Gorla & Lam [20], Bradley & Hebert [30], and Cunha & Greathead [31] used MBTI in the area of Information Systems and Software Engineering. An individual's personality type in MBTI is assessed in four dimensions: social interaction (extroversion (E) and introversion (I)), decision making (thinking (T) and feeling (F)), infor-

mation gathering (sensing (S) and intuition (N)), and dealing with the external world (judging (J) and perceiving (P)) [7]. The MBTI test allows individual personality type preferences to be classified according to the 16 types that results as a combination of four dimensional pairs, which are Introversion (I) and Extroversion (E); Thinking (T) and Feeling (F); Sensing (S) and Intuitive (N) ; and Judging (J) and Perceiving (P). The 16 possible personality combinations are formed from four dimensions shown in the following Table 1.

Table 1. Keywords for search.

ISTJ (1)	ISFJ (2)	INFJ (3)	INTJ (4)
ISTP (5)	ISFP (6)	INFP (7)	INTP (8)
ESTP (9)	ESFP (10)	ENFP (11)	ENTP (12)
ESTJ (13)	ESFJ (14)	ENFJ (15)	ENTJ (16)

3. MAJOR ROLES IN SOFTWARE DEVELOPMENT TEAMS

This study employs the term 'role' as "the function assumed or part performed by a person sometimes in a particular situation". A team member performs numerous chores that reveal his/her different roles at a time. For instance, a team leader can also be entitled as a tester or a system analyst in software engineering. Moreover, the following sub sections highlight the different roles of software engineers performed during software development process at different times.

3.1 System Analyst

The system analyst has the responsibilities to gain an access to the requirements raised by the client, to meet the needs of client, to develop the awareness pertinent to the key characteristics of system, and to merge these all requirements into application model so as to address the client's needs. Additionally, the system analysis is reckoned as the most important and critical phases of all in software development phases where real world components of applications are determined and modules are formed for composing the system [32].

3.2 Software Designer

In the beginning, during software designing phase of system life cycle of software, designer searches for the components of the system for addressing the problem revealed by the clients. In the same phase, complex tasks are composed separately. Although work related to this phase seems an easy to handle with but in fact it requires well equipped adroitness to address the needs of key components [33].

3.3 Programmer

In the software development process, programmer has the key position for implementing the designing phase of a system. The sensitivity of programmer's role lie in a fact that the programmer must be adept in syntax of the programming and good at ana-

lytical and logical sharpness for finding the code of the program with an ease. The lack of these qualities could make programmer face terrible failure because programming phase has the crucial importance which is used to apply and identify data structures, control structure of the program and determines relevant variables [34, 35].

3.4 Tester

Tester has also undeniable importance in the process of the software development where he/she carries responsibility to ensure the smooth run of the system. So, testing is equally important which determines the defects dulling the software. Moreover, testing does not have any phase to start from but it functions right from the beginning to ensure the software development process is rightly on the track. Testing also involves many techniques such as system testing, integration and unit. Additionally, some exceptions traced in the system are termed as a bug and process to resolve the bugs is called debugging [36].

3.5 Team Leader

The position of a team leader can be defined like the role of the orchestra leader who guides his team members of a music band. Team leader supervises and monitors the whole process of the development of the software at all the phases. Team leader does not only strive to bring success to the software but he/she is also responsible to keep higher authorities updated by providing them reports on software. Moreover, a leader also supervises different projects to ensure their quality [37].

4. METHODOLOGY

The main purpose of this research paper was to find the suitable personality types of programmer with the team-leader and programmers by gender classification in software development teams. In other words, it could also be said that this research paper tried to find the effective personality types' equations between team-leader and programmer in software development by focusing gender as an impacting variable. Keeping in view, member-role, gender, and personality type of individual were considered as the predictor variables of this research. In addition, the impact of these predictor variables were measured on an outcome variable *i.e.*, team-performance. Following Table 2 shows the variables and possible inputs used in this study.

In order to find the possible combination of personality types between team-leader and programmer, this study applied predictive approach (*i.e.*, rough sets) on the data. In the year of 2015, students who were learning software engineering subject during their bachelor from Universiti Utara Malaysia (UUM) participated voluntarily in the experimental process of data collection. The software engineering class was chosen with two reasons: age and programming subject experience. This study required only those students whose age are either 20 or above because 20 years is an age where personality gets stable [38]. Therefore, the results emanating from study may be generalized with industry of the population country *i.e.*, Malaysia. Moreover, second condition was to confirm the technical skills of the participants. Hence, students who were learning software engineering have already taken several programming subjects: structured programming,

object oriented programming, and web programming subjects. Because, the participated students were supposed to develop a programming based project in groups, it is worth mentioning here that this study focused personality equations of participant than the technical skills of them.

First of all, students were grouped in several teams with one team leader and other as programmers. Team leader was responsible to receive the project requirements from subject teacher, distribute the tasks among members, and to get and bring the updates of project to the class teacher. Furthermore, agile development methodology was followed by the students during development of projects. Because, small-medium teams were composed with 4-6 members for short period of time, as it is one of the attractiveness of the agile development methodology that it is suitable for small-medium teams for small development projects. Students were asked to submit the projects with line of code, and documentation of the projects was not required. Moreover, the progress of groups were observed and monitored to check the quality points of the projects. Lastly, those all projects were submitted to the class teacher within 14 weeks. These projects were then evaluated by requirement engineer to see their quality of development in the light of projects' requirement. For example, all submitted projects were assigned certain marks based on the development quality. Those teams were considered effective if the submitted project obtained 80% or above. In the same way, teams were named ineffective if their submitted project could not obtain 80% marks.

Table 2. List of Variables used in this research.

Variable	Input
Predictor	
1. Member role	Team-leader
	Programmer
2. Personality types	**16 types of MBTI** 1 = ISTJ 2 = ISFJ 3 = INFJ 4 = INTJ 5 = ISTP 6 = ISFP 7 = INFP 8 = INTP 9 = ESTP 10 = ESFP 11 = ENFP 12 = ENTP 13 = ESTJ 14 = ESFJ 15 = ENFJ 16 ENTJ
3. Gender	1=Male
	2=Female
Outcome	
4. Team performance	Quality(0)=Ineffective
	Quality(1)=Effective

Total 184 students participated in the experiments of project development in 45 teams. In which, 17 projects were declared as effective-projects and 28 were said ineffective-projects by requirement engineer (*i.e.*, based on the requirements). Importantly, this research only focused the successful software teams for finding efficient personality types' equations for team-leader and programmer. Therefore, only 68 members' personality type counted from those 17 effective-projects. Moreover, 17 team-leaders were counted (*i.e.*, each team one leader) and 51 programmers appeared in those shortlisted projects. It should also be noted here that this paper employed the term "effective" to refer results obtained from those projects which met the requirements and other way around for term "ineffective".

Apart from statistical information, predictive approach of rough sets was applied on the overall data by following steps:

1 Data was split into two major and basic sets: training and testing, with 70% and 30% standard ratio [39-41].
2 Genetic Algorithm (GA) was applied on training set by using ROSETTA toolkit for finding possible equations (or rules) of personalities.
3 Standard Voting (SV), Voting with Object tracking, and Naïve Bayes classification techniques were then applied on the results emanated from GA algorithm (or 2nd step) for finding prediction accuracy.
4 Filtering only effective results for suggesting possible effective personality types for team-leader and programmer.

5. RESULTS AND DISCUSSION

This section discusses the results in two main sections: Programmer personality types with Team-leader personality types; and Programmer personality types with Programmer personality types. The first part thoroughly discusses the possible equations of programmers' personality type with the team-leader personality types while second part covers the suitable personality types among different programmers in the same team.

5.1 Programmer Personality Types with Team-Leader Personality Types

To extract the reliable results, GA was not the only algorithm which was applied but Johnson's Algorithm (JA) was also applied to extract the trends from training set. Eventually, the GA results emerged as efficient with the objective of the research. Moreover, the application GA algorithm extracted 60 rules from training data set. Among these 60 rules, 22 rules were shortlisted for further discussion and the rest 38 rules were dropped for showing ineffective endings in results. Following Table 3 is showing the efficient rules for team-leader and programmer composition in team.

Prior to the discussion of the results, it is important to discuss the structure of the rules presented in Table 3 above. First of all, the rules (or personality-type equations) are divided into two parts: condition and outcome. For instance, the side before "⇒" sign is highlighting the "IF" part of the equation and the part after "⇒" is "THEN" part. It is, therefore, the IF part could also be called Left Hand Side (LHS) and the THEN part as Right Hand Side (RHS). Table 3 possessed two types of rules for the formation of pro-

grammer with team-leader. The first type of rules did not discuss the gender classification of programmer but they only discuss the personality types of team-leader and programmer. Whereas, the second type of rules were composed of team-leader, programmer personality types, and gender of programmer. Technically, both types of rules were important to consider for programmer personality types' composition with team-leader since the dataset was small and it would not have covered the overall range of personality types' possibilities. Therefore, this research measured that the composition of programmer with team-leader is generic for both genders if the gender is not mentioned. It means that the equation for programmer without gender classification can be used to compose team-leader with any type of gender programmer.

Table 3. List of variables used in this research.

Rule No	Rules/Equations
1	Leader_Personality(ENTJ) AND Programmer_Personailty(ENTJ)⇒No_Quality OR Quality
2	Leader_Personality(ENFJ) AND Programmer_Personailty(ESTJ)⇒Quality
3	Leader_Personality(ENFJ) AND Programmer_Personailty(ENFJ)⇒Quality
4	Leader_Personality(ENTJ) AND Programmer_Personailty(ISFJ)⇒Quality
5	Leader_Personality(ENFJ) AND Programmer_Personailty(ISTP)⇒Quality
6	Leader_Personality(INFJ) AND Programmer_Personailty(INTJ)⇒Quality
7	Leader_Personality(INFJ) AND Programmer_Personailty(ENTJ)⇒Quality
8	Leader_Personality(INFJ) AND Programmer_Personailty(ENFJ)⇒Quality
9	Leader_Personality(ISTJ) AND Programmer_Personailty(ENTJ)⇒Quality
10	Leader_Personality(INFP) AND Programmer_Personailty(ISTJ) ⇒ Quality
11	Leader_Personality(ENFJ) AND Programmer_Personailty(ENFP)⇒Quality
12	Leader_Personality(ISFJ) AND Programmer_Personailty(INTJ)⇒Quality
13	Leader_Personality(ISFJ) AND Programmer_Personailty(ISFP)⇒Quality
14	Leader_Personality(ESTJ) AND Programmer_Personailty(ENFP)⇒Quality
15	Leader_Personality(ESTJ) AND Programmer_Personailty(ENFJ)⇒Quality
16	Leader_Personality(INFJ) AND Programmer_Personailty(ISTJ)⇒Quality
17	Leader_Personality(ENTJ) AND Programmer_Personailty(ESTJ)⇒Quality
18	Leader_Personality(ENTJ) AND (Programmer_Personailty(ENFJ) ANDFemale)⇒Quality OR No_Quality
19	Leader_Personality(ENTJ) AND (Programmer_Personailty(ENFJ) AND Male)⇒Quality
20	Leader_Personality(INFP) AND (Programmer_Personailty(ENFJ) AND Male)⇒Quality
21	Leader_Personality(ESFJ) AND (Programmer_Personailty(ESFJ) AND Male)⇒Quality
22	Leader_Personality(ESTJ) AND (Programmer_Personailty(ESFJ) ANDFemale)⇒Quality

Based on the results presented in Table 3, the first 17 rules were sorted as a generic rule for both gender programmers (*i.e.*, male or female) with team-leader. In which, team-leader with personality type number 13, 15, and 16 (or EJ (Extrovert and Judging) combination personality type leader) were adjusted with E trait programmers. In other words, E trait programmers were more convenient with the leaders of E trait. For instance, personality type number 13, 15, and 16 appeared as team-leader for 9 programmers (see Table 3 rule number 1, 2, 3, 4, 5, 11, 14, 15, and 17). In which, majority of

programmers were E trait (see Table 3 rule number 1, 2, 3, 11, 14, 15, and 17) and two rules remained efficient with I (Introvert) trait but with S (Sensing) trait. Because, a team leader has responsibility to communicate frequently with his team members and management [42]. Whilst, extrovert (E) people like those who also response them back in a good way. Therefore, it may be a reason that E programmers were found effective with E trait leaders. In the same vein, team-leader with IJ (introvert and Judging) combination was found moderated with I and E traits programmers. As the rule number 6, 7, 8, 9, and 16 from Table 3 showed that programmer with I and J traits can also work well with team-leaders possessing I and J traits (see rule number 6 and 16 from Table 3 above). Moreover, the rule number 10, 12, and 13 showed programmers and leaders with I trait personality and their results appeared quite efficient. Therefore, one can suggest that I trait programmer can work happily with I trait team-leaders provided S trait is in the personality types of programmer or the team has equal number of members from both I and E traits. Yet, previous studies have not proposed any leader and programmer combination results but they at least have suggested the I trait for programmer role [7, 43, 44]. These studies highlight that programming job is a cognitive and problem solving job. They further believe that it has most to do with analytical reasoning than the interactions.

On the other hand, 5 rules were combination of team-leader + (programmer+gender) personality types. These rules were richer than the previous discussed rules because they include the gender of programmers with team-leader personality types. Keeping in view, rule number 18 clearly says that if the personality type is ENFJ and the gender is female with ENTJ team-leader then they may create 60% efficient outcomes. But, the efficient outcomes can be ensured on a condition when male programmer work together having similar personality types (see rule number 19 above). Similarly, ESFJ (personality type number 14) is suitable for male programmer for the same type of personality team-leader. But, it does not assure at any level that the female programmer with ESFJ personality type will create equation with ESFJ team-leader or not. However, female programmer with ESFJ can be efficient if team-leader's personality T trait is exchanged with F trait. It means that ESFJ female programmer can produced efficient results if the team-leader is ESTF. Additionally, literature has not emphasized the ESFJ personality for programmer yet [44, 45]. But, this study has found it effective with gender and leader personality conditions. This can happen because the population of this study is different. On the other hand, it is also believed that no personality is good or bad but suitability matters. Lastly, N and F traits were found fruitful only if there was male gender and the leader possessing I and P traits.

5.2 Validity of the Rules

After extracting the rules to determine the effective programmer for team from the data by using GA algorithm technique, it seemed an appropriate to find their validity. For this purpose, testing dataset was used with Standard voting, voting with object tacking, and Naïve Bayes classifiers on the results extracted from training set (extracted rules). Among them, Naïve Bayes produced very high predication accuracy with 78.57% whereas 71.42% and 69.04% were obtained from Standard Voting and Voting with Object Tracking respectively.

Table 4. Results of prediction accuracy.

Classifier	Accuracy
Naïve Bayes	78.57%
Standard Voting	71.42%
Voting with Object Tracking	69.04%

Moreover, many researchers have benched mark their study with 70% accuracy. For instance, Bakar [46] and Hvidsten [47] stated that results will be acceptable and effective if the prediction accuracy reaches 70% or above. Hence, this study also sets 70% as the benchmark of effectiveness. Based on it, only voting with object tracking couldn't generate the benchmark accuracy but Naïve Bayes and Standard Voting obtained the benchmark level. Therefore, rules proposed from this study can be used for finding effective programmer with team-leader and other programmers within the team.

5.3 Programmer Personality Types with Programmer Personality Types

The first part of the discussion was based on the equations between team-leader and programmers' personality types. The gender type of programmer was also highlighted to gain the deep understanding of the personality of programmer role. Additionally, in this section, the programmers' personality types are discussed with the other programmers' personality types within the same team. The following Table 5 depicts the effective personality types among programmers and their gender composition.

Table 5. Efficient traits based on Leader personality types.

Team No	Programmers Personality types' Composition	Gender Composition
1	ISTJ,ENTJ,ENTJ	FFF
2	ISTJ,INTJ,ISFP	FFF
3	INTJ,ENFJ,ENTJ	MMM
4	ISFJ,INFJ,INTJ,ENFJ	MFFF
5	ISTJ,INFJ	FF
6	ISTJ,ENFJ,ENTJ	MMM
7	ENFP,ENFP,ENFP	MFF
8	ISFP,ESFJ,ENFJ	FFF
9	ISFJ,ESFJ,ESFJ	FFM
10	ESTJ,ENFJ,ENFJ	FFF
11	ESTJ,ESTJ,ESFJ	MMM
12	ISTP,ESTJ,ESFJ	MFM
13	ISTJ,ISFP,ENFP	FFM
14	ISTJ,ISFJ,ESTJ,ENFJ	FFFF
15	ENFJ,ENTJ,ENTJ,ENTJ	MMMM
16	ISTJ,ESTJ	FF
17	ENFJ,ENFJ	FF

It is mentioned in the research methodology section above, only 17 teams were able to obtain the successful requirements of the projects. Therefore, Table 5 presents the data of all those teams which could manage to produce the projects with quality. Moreover, the first discussion part of this research highlighted, based on this dataset, possible effec-

tive equations between team-leader and programmers' personality types. It is worth mentioning here that programmer does not work with team-leader in a team. Programmers often work with other programmers to achieve the goal of project in the same team. Hence, it is equally important to know the equations between programmers within the same team.

First of all, based on the results of Table 5 given above, this research found strong evidence that the programmers' gender homogeneity can lead the project towards success. For instance, among 17 successful teams, solely 12 teams of homogeneous gender programmers emerged successful within the dataset. Interestingly, majority male programmer personality types showed E traits for which female programmers were quite opposite to possess diverse personality types. Fig. 1 shows the percentage of male and female gender diversity in the successful teams.

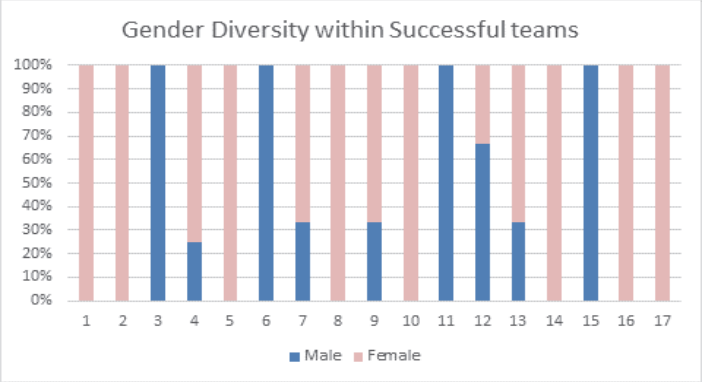

Fig. 1. Programmers' gender diversity within the successful teams.

Data from Table 5 was further categorized in three groups: only female programmers, only male programmer, and mix gender programmers. In which teams 1, 2, 5, 8, 10, 14, 16, and 17 (see Table 5) were the programmers' composition consisting only female gender. On another hand, team number 3, 6, 11, and 15 (see Table 5) from successful teams were based on only male gender programmers. But, number 4, 7, 9, 12, and 13 were the successful teams composed of diverse genders with different personality compositions. Keeping the mix gender teams in view, it was found that if the team was composed of majority of female programmer members then MBTI personality type number 2, 11, and 14 were the better choices for accommodating new or existing members. Whereas, only 13 number of MBTI personality type females were adjustable with the majority male programmers' teams.

6. THREATS TO VALIDITY

First of all, the results emanated from this study can be used for finding effective personality composition for programmers but personality is a complex term which is vague in nature and can be impacted from several internal and external factors. Thus, the generalization of the results remain the main concern of the validity. For instance, results

from this study cannot be generalized with other Malaysian universities without cross validation. In order to generalize them, the results can be expanded with other universities of Malaysia and as well with multicultural settings for more rules. Secondly, the study experiments were conducted within academic setups which limits the results for industrial settings. To validate the results, data from industrial settings can remove the threats of acceptance or generalization. Thirdly, team-lead and programmer roles were adjusted to compose personality equations for effective programmer. Usually, programmer also work with tester and designers. In this case, the results are restricted to be used within the teams which involve roles other than team-lead and programmers. Thus, the results can be enriched if these are trend with other roles: testers or designers. Furthermore, only MBTI based personality compositions are offered in the model which can be one of the limitations. Hence, the results can include the new rules based on personality assessment other than MBTI: Big Five or Keirsey Temperament Sorter.

7. CONCLUSION

This research drew the pivotal importance of human aspect issues within the software development workplaces. It is considered one of the seven factors for software development failure in workplaces. Moreover, this research participated in software development literature by giving equations for finding effective programmer for team composition. For that purpose, academic population was focused to simulate the development environment. The experiments started from data distribution with 70%-30% of training and testing sets. GA algorithm's results were chosen for further discussion and testing because it produced more effective results than JA algorithm. The key significance of the findings of this research lies in a fact that programmer's role was explained from different angles. Such as, E trait team-leader were found more efficient with E trait programmers than others while I trait team-leader appeared convenient working with I and E traits programmers. The results also suggested that I trait programmers were easy with team-leaders possessing I traits provided programmers possessed S trait of personality. Additionally, the findings also suggested that ENFJ personality male programmer rendered acute efficiency working with ENTJ personality team-leader. On the other hand, ESJF personality type emerged as an ideal mean for female programmer if team-leader was ESTF. Moreover, this research also validated the results with few classification techniques. In which, Naïve Bayes obtained higher accuracy with 78.57% where the benchmark was only 70%. The standard voting, other classification technique, also obtained 71.42% prediction accuracy. Thus, the determination of the validity of the results through different classification techniques suggested practitioners to implement them for finding effective programmers. This research further detailed the programmer role with the other programmers within the teams. The gender of programmer with personality types was considered to obtain the results from data. Only descriptive analysis was applied to highlight the impacting results were from the dataset. The findings were significant due to some reasons. Firstly, the findings asserted that same-gender programmers within the teams were more productive than diverse-gender programmers. Secondly, results also confirmed that personality type ISFJ, ENFP, and ESFJ are suitable options for female majority teams. Lastly, only ESTJ personality type females were found adjustable within the male majority teams.

8. FUTURE WORK AND RECOMMENDATION

This study provides some valuable suggestions for researchers who wish to contribute within this scope of software development research. First, the future work can include large data that can generate more conducive patterns for team composition. Second, the present study employed certain techniques of data mining to suggest the model for programmer's role; therefore, future work may be undertaken by using other techniques to establish a better accuracy in the results. Third, since this study solely accentuated on the programmers' role in software development, thus someone can also include other roles of software development to extend the scope of the model. Last but not the least, industrial settings are very much important to see the real impact of these types of model. Hence, it is also recommended for future to involve industrial setting's data to make the findings of the model more generalizable and applicable without any ambiguities.

REFERENCES

1. B. Wysocki Jr, "Some firms, let down by costly computers, opt to 'de-engineer,'" *Wall Street Journal*, Vol. 150, 1998, pp. 1-2.
2. T. Big and A. Small, "Chaos Manifesto 2013," 2013.
3. R. L. Glass, "Short-term and long-term remedies for runaway projects," *Communications of the ACM*, Vol. 41, 1998, pp. 13-15.
4. M. Keil, A. Rai, J. E. Cheney Mann, and G. P. Zhang, "Why software projects escalate: The importance of project management constructs," *IEEE Transactions on Engineering Management*, Vol. 50, 2003, pp. 251-261.
5. B. Bang, "Big bang boom," 2014, pp. 1-12.
6. A. R. Gilal, M. Omar, and K. I. Sharif, "A rule-based approach for discovering effective software team composition," *Journal of Information and Communication Technology*, 2014, pp. 1-20.
7. L. F. Capretz and F. Ahmed, "Making sense of software development and personality types," *IT Professional*, Vol. 12, 2010, pp. 6-13.
8. T. Dingsøyr and T. Dybå, "Team effectiveness in software development human and cooperative aspects in team effectiveness models and priorities for future studies," in *Proceedings of the 5th International Workshop on Co-operative and Human Aspects of Software Engineering*, 2012, pp. 27-29.
9. J. E. Stake and H. Eisele, "Gender and personality," *Handbook of Gender Research in Psychology*, J. C. Chrisler, D. R. McCreary, eds., Chapter 2, 2010, pp. 19-40.
10. A. R. Gilal, J. Jaafar, S. Basri, M. Omar, and M. Z. Tunio, "Making programmer suitable for team-leader: Software team composition based on personality types," in *Proceedings of International Symposium on Mathematical Sciences and Computing Research*, 2015, pp. 78-82.
11. D. Robey and M. Newman, "Sequential patterns in information systems development: an application of a social process model," *ACM Transactions on Information Systems*, Vol. 14, 1996, pp. 30-63.
12. S. Sawyer, "Effects of intra-group conflict on packaged software development team performance," *Information Systems Journal*, Vol. 11, 2001, pp. 155-178.

13. W. Scacchi, "Socio-technical interaction networks in free/open source software development processes," in *Software Process Modeling*, Springer, 2005, pp. 1-27.
14. D. Robey, "Research report–modeling interpersonal processes during system development: Further thoughts and suggestions," *Information Systems Research*, Vol. 5, 1994, pp. 439-445.
15. S. Sawyer and P. J. Guinan, "Software development: Processes and performance," *IBM Systems Journal*, Vol. 37, 1998, pp. 552-569.
16. L. S. Kirsch, "Portfolios of control modes and IS project management," *Information Systems Research*, Vol. 8, 1997, pp. 215-239.
17. B. Curtis, H. Krasner, and N. Iscoe, "A field study of the software design process for large systems," *Communications of the ACM*, Vol. 31, 1988, pp. 1268-1287.
18. E. M. Trauth, "Theorizing gender and information technology research," *Encyclopedia of Gender and Information Technology*, Vol. 2, 2006, pp. 1154-1159.
19. A. R. Gilal, J. Jaafar, M. Omar, and M. Z. Tunio, "Impact of personality and gender diversity on software development teams' performance," in *Proceedings of IEEE International Conference on Computer*, *Communication*, *and Control Technology*, No. 2014, 2014, pp. 261-265.
20. N. Gorla and Y. W. Lam, "Who should work with whom?: building effective software project teams," *Communications of the ACM*, Vol. 47, 2004, pp. 79-82.
21. L. F. Capretz and F. Ahmed, "Why do we need personality diversity in software engineering?" *ACM SIGSOFT Software Engineering Notes*, Vol. 35, 2010, p. 1.
22. S. S. J. O. Cruz, F. Q. B. da Silva, C. V. F. Monteiro, C. F. Santos, and M. T. dos Santos, "Personality in software engineering: preliminary findings from a systematic literature review," in *Proceedings of the 15th Annual Conference on Evaluation and Assessment in Software Engineering*, 2011, pp. 1-10.
23. L. McLeod and S. G. MacDonell, "Factors that affect software systems development project outcomes," *ACM Computing Surveys*, Vol. 43, 2011, pp. 1-56.
24. M. Omar, S.-L. Syed-Abdullah, and N. M. Hussin, "Analyzing personality types to predict team performance," in *Proceedings of International Conference on Science and Social Research*, 2010, pp. 624-628.
25. J. S. Karn, S. Syed-Abdullah, A. J. Cowling, and M. Holcombe, "A study into the effects of personality type and methodology on cohesion in software engineering teams," *Behaviour and Information Technology*, Vol. 26, 2007, pp. 99-111.
26. J. Karn and T. Cowling, "A follow up study of the effect of personality on the performance of software engineering teams," in *Proceedings of ACM/IEEE International Symposium on Empirical Software Engineering*, 2006, pp. 232–241.
27. A. R. Gilal, M. Omar, and K. I. Sharif, "Discovering personality types and diversity based on software team roles," in *Proceedings of International Conference on Computing and Informatics*, 2013, pp. 259-264.
28. M. Omar, S.-L. Syed-Abdullah, and N. M. Hussin, "Developing a team performance prediction model: A rough sets approach," *Informatics Engineering and Information Science*, 2011, pp. 691-705.
29. A. Furnham, "The big five versus the big four: the relationship between the Myers-Briggs Type Indicator (MBTI) and NEO-PI five factor model of personality," *Personality and Individual Differences*, Vol. 21, 1996, pp. 303-307.

30. J. H. Bradley and F. J. Hebert, "The effect of personality type on team performance," *Journal of Management Development*, Vol. 16, 1997, pp. 337-353.
31. A. D. Da Cunha and D. Greathead, "Does personality matter?: an analysis of code-review ability," *Communications of the ACM*, Vol. 50, 2007, pp. 109-112.
32. G. Shelly and M. Vermaat, *Discovering Computers 2009: Introductory*, Cengage Learning, MA, 2008.
33. M. Gualtieri and J. Darrow, *Architecture Complete: A Programmer's Guide to Software Architecture and Design*, Wrox Press, Birmingham, 2012.
34. S. Licorish, A. Philpott, and S. G. MacDonell, "Supporting agile team composition: A prototype tool for identifying personality (in) compatibilities," in *Proceedings of Workshop on Cooperative and Human Aspects on Software Engineering*, 2009, pp. 66-73.
35. A. R. Gilal, J. Jaafar, S. Basri, M. Omar, and A. Abro, "Impact of software team composition methodology on the personality preferences of Malaysian students," in *Proceedings of the 3rd International Conference on Computer and Information Sciences*, 2016, pp. 454-458.
36. G. J. Myers, C. Sandler, and T. Badgett, *The Art of Software Testing*, John Wiley & Sons, NJ, 2011.
37. S. Sarin and C. McDermott, "The effect of team leader characteristics on learning, knowledge application, and performance of cross-functional new product development teams," *Decision Sciences*, Vol. 34, 2003, pp. 707-739.
38. P. T. Costa Jr and R. R. McCrae, "Personality stability and its implications for clinical psychology," *Clinical Psychology Review*, Vol. 6, 1986, pp. 407-423.
39. C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Transactions on Neural Networks*, Vol. 13, 2002, pp. 415-425.
40. J. W. Grzymala-Busse, "A new version of the rule induction system LERS," *Fundamenta Informaticae*, Vol. 31, 1997, pp. 27-39.
41. R. Menon, G. Bhat, G. R. Saade, and H. Spratt, "Multivariate adaptive regression splines analysis to predict biomarkers of spontaneous preterm birth," *Acta Obstetricia et Gynecologica Scandinavica*, Vol. 93, 2014, pp. 382-391.
42. A. R. Gilal, J. Jaafar, M. Omar, S. Basri, and A. Waqas, "A rule-based model for software development team composition: Team leader role with personality types and gender classification," *Information and Software Technology*, Vol. 74, 2016, pp. 105-113.
43. J. Teague, "Personality type, career preference and implications for computer science recruitment and teaching Joy Teague School of Management Information Systems Waurn Ponds Campus Deakin University, 3217," *Journal of Management Information Systems*, 1998, pp. 155-164.
44. L. F. Capretz, "Psychological types of Brazilian software engineering students," *Journal of Psychological Type*, 2008, pp. 37-42.
45. D. Varona, L. F. Capretz, and Y. Piñero, "Personality types of Cuban software developers," *Global Journal of Engineering Education*, Vol. 13, 2011, pp. 77-81.
46. A. B. Nauman, R. Aziz, A. F. M. Ishaq, and M. Mohsin, "An analysis of capabilities of Pakistan as an offshore IT services outsourcing destination," in *Proceedings of the 8th International Multitopic Conference*, 2004, pp. 403-408.

47. T. R. en Hvidsten, "Fault diagnosis in rotating machinery using rough set theory and ROSETTA," Department of Computer and Information Science, Norwegian University of Science and Technology, 1999.

Abdul Rehman Gilal is a Ph.D. candidate in the Department of Computer and Information Sciences at Universiti Teknologi Petronas, Malaysia. He holds a degree of MSc (IT) from Universiti Utara Malaysia, 2013. His areas of interest are human aspect in software development, complex networks, databases and data mining, programming and cloud computing.

Jafreezal Jaafar is an Associate Professor and former Head of CIS at Universiti Teknologi PETRONAS, Malaysia. He holds a PhD degree from University of Edinburgh, UK, 2009. He did MAppSc in IT from RMIT University, Australia, 2002 and bachelor's degree in Computer Science from Universiti Teknologi Malaysia, 1998. His main research areas include big data analytics, soft computing and software engineering. He has managed to secure several number of research projects from industry and government agencies. Based on his publication tract record, he had been appointed as a Chief Editor and reviewer for several journals. He also possesses a great experience for organizing and participating as a technical chair or member for several international conferences. He was appointed as an Executive Committee member for IEEE-Computer society, Malaysia-chapter in 2016 and 2017.

Ahsanullah Abro has received his Bachelor degree in Computer Science from Shah Abdul Latif University Khairpur Mir's, Sindh, Pakistan, 2006. MSc IT (by research) from UTP, Malaysia, 2011. Ahsanullah Abro is currently pursuing his Ph.D. studies in IT at UTP, Malaysia. Previously, he has worked as a Research Associate and an Assistant Professor in Public Institutes of higher learning and participated in various academic activities and research projects. He teaches various engineering and ICT courses to undergraduate students. His research interest includes design science, development of engineering and computer science applications, and User Experience (UX) evaluation of systems, products and services.

Waheed Ali Umrani works as an Assistant Professor, at the Business School, Sukkur Institute of Business Administration, Sukkur, Sindh, Pakistan. Umrani holds Ph.D. in Management Sciences from University Utara Malaysia and an MBA from Pakistan. His research interest includes Leadership, Corporate Entrepreneurship, Employee Engagement, and Service Quality. Dr. Umrani is founder and is currently leading Sukkur IBA Case Research Center, an initiative to promote case teaching and case research at Sukkur IBA. As a trainer Umrani has conducted professional trainings for corporate and academicians in Malaysia, United Arab Emirates, and all around Pakistan on data-analysis and research methods and leadership development. Umrani has published in Business and Economics Journal, Emerald Emerging Case Study Journal, International Postgraduate Business Journal, Mediterranean Journal of Social Sciences, and Management Science Letters. His work has also been accepted for publication in Iranian Journal of Management studies and PSU Research Review.

Shuib Basri was born in Kedah, Malaysia, in 1972. He received the BS in IT with Hons from University Utara Malaysia, the MSSE from Melbourne University, Australia in 2001, and Ph.D. degrees in Software Engineering from the Dublin City University, Republic of Ireland in 2010. In 2000, he joined the Department of CIS, University Teknologi PETRONAS, as a tutor and aster obtain his Master he work as a Lecturer, until now. His current research interests include software quality, software matrix, software process improvement and also research related to big data such as data synchronization and categorization. Shuib Basri is a member of IEEE Malaysia and IEEE computer Society Malaysia. Work closely with several industry *e.g.* Microsoft, Malaysia Testing Board and Malaysia Quality Board.

Mazni Omar graduated with honours degree in Information Technology from Universiti Utara Malaysia in 2000. She received MSc degree in Software Engineering from Universiti Teknologi Malaysia in 2002. In 2012, she received the Ph.D. degree from Universiti Teknologi MARA, Malaysia for a thesis on the empirical studies of agile methodology in humanistic aspects. She began her career as a tutor in School of Information Technology, UUM in 2000. After getting her Master degree, she was appointed as lecturer at the same school in 2002. Then, after getting her Ph.D., she is appointed as Senior Lecturer at School of Computing. Her field of expertise is in software engineering that focuses on human aspects, as well as on data mining and knowledge management. Her field of interest covers across a multi-disciplinary field of computing and social sciences applications to the human aspects of software engineering education.