arXiv:1603.00696v2 [cs.SE] 3 Mar 2016

Finding Relationships between Socio-technical Aspects and Personality Traits by Mining Developer E-mails

Oscar Hernán Paruma-Pabón [1]
ohparumap@unal.edu.co

Fabio A. González [1]
fagonzalezo@unal.edu.co

Jairo Aponte [1]
jhapontem@unal.edu.co

Jorge E. Camargo [2]
jorgecamargo@uan.edu.co

Felipe Restrepo-Calle [1]
ferestrepoca@unal.edu.co

[1] MindLab research group, Universidad Nacional de Colombia, Bogotá, Colombia

[2] Lacser research group, Universidad Antonio Nariño, Bogotá, Colombia

ABSTRACT

Personality traits influence most, if not all, of the human activities, from those as natural as the way people walk, talk, dress and write to those most complex as the way they interact with others. Most importantly, personality influences the way people make decisions including, in the case of developers, the criteria they consider when selecting a software project they want to participate. Most of the works that study the influence of social, technical and human factors in software development projects have been focused on the impact of communications in software quality. For instance, on identifying predictors to detect files that may contain bugs before releasing an enhanced version of a software product. Only a few of these works focus on the analysis of personality traits of developers with commit permissions (committers) in Free/Libre and Open-Source Software projects and their relationship with the software artifacts they interact with. This paper presents an approach, based on the automatic recognition of personality traits from e-mails sent by committers in FLOSS projects, to uncover relationships between the social and technical aspects that occur during the software development process. Our experimental results suggest the existence of some relationships among personality traits projected by the committers through their e-mails and the social (communication) and technical activities they undertake. This work is a preliminary study aimed at supporting the setting up of efficient work teams in software development projects based on an appropriate mix of stakeholders taking into account their personality traits.

CCS Concepts

•**Software and its engineering** → **Programming teams;** •**Information systems** → *Open source software; Data mining; Clustering;* •**Human-centered computing** → *Open source software;*

Keywords

socio-technical aspects, FLOSS, personality traits, source code repository, mailing lists, email communication.

1. INTRODUCTION

The main motivation of this paper is to study the relationships between software artifacts, mainly source code, and developers' personality traits to gain insight into the factors influencing developers to write code for any or some modules in a FLOSS project. It is expected these insights may help to explain the implicit mechanisms that lead to self-forming software teams, and how the developers' personality marks are reflected in some technical actions such as frequency of the commits, size of the sent patches, and terms used in commit messages.

In the present work, we are looking for relationships between social and technical aspects in the evolution of FLOSS projects, seeking to answer the following research questions:

- what personality traits can be identified through communications among software developers involved in FLOSS projects?,
- what personality traits stand out according to the projects the software developers are involved in?, and
- what relationships can be observed between the social activities (communication through the project mailing lists) of the committers and personality traits characterizing the groups they belong to?

To the best of our knowledge, this paper is one of the first analyzing personality traits of software developers in FLOSS projects and their relationships with social and technical aspects.

The paper is structured as follows: In Section 2 we present a review of the related work; in Section 3 we describe the methodology of the study; Section 4 describes the experiments conducted and the results obtained; Section 5 presents

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

CHASE'16, May 16 2016, Austin, TX, USA
© 2016 ACM. ISBN 978-1-4503-4155-4/16/05…$15.00
DOI: http://dx.doi.org/10.1145/2897586.2897611

the conclusions, and lastly, we discuss threats to validity in Section 6.

2. RELATED WORK

Some authors agree that most of the research into the software development process has been focused on technical aspects [4, 5]. However, some studies have addressed the role of human factors, e.g. personality traits, in software engineering and software development. Sheppard and Curtis[2] report results from experiments to determine the influence of human factors in software development.

Basili and Reiter[3] remark that factors directly related to the psychological nature of human beings play a major role in software development. They concluded that research into the effects of human factors on software is dependent on suitable measurement of several non-functional software features. They report findings indicating that a larger programming team size and the use of a disciplined methodology have beneficial effects on the development process and the developed product.

In their review of productivity factors in software development, Wagner and Ruhe[4] give a special consideration of human factors in software engineering. Such factors, as they explain, are often not analyzed with equal detail as more technical factors further than more than a third of the time a software developer is concerned with other kind of work, not just technical work. One of the main contribution of Wagner and Ruhe's work is the list of soft and technical factors influencing productivity in software development they provided.

The work of Sommerville and Rodden[5] discusses human, social, and organizational factors affecting software processes and, to remark, they discuss how to analyze software processes as human rather than technical processes.

With regard to software process enhancements, Acuña and Juristo[13] proposed a Capabilities-oriented Software Process Model for assigning people to roles according to their capabilities and the capabilities demanded by the role, and empirically validated its positive impact in software development effectiveness and efficiency. Along the same line of work, seeking to associate personality with the software process, Bradley and Hebert[14] proposed a model that can be used to analyze the personality type composition of an information system development team and highlighted the impact of personality type on team productivity. Capretz[17] provides a personality profile of software engineers according to the Myers–Briggs Type Indicator and the results of his study suggests that software engineers are most likely to be ST (Sensing and Thinking) or TJ (Thinking and Judgment) or NT (Intuition and Thinking). Most recently, Capretz and Ahmed[16] used the Myers-Briggs Type Indicator (MBTI), a self-inventory designed to identify an individual personality type, strengths, and preferences, to mapping job and skills requirements to personality types for each of the activities involved in software engineering processes such as system analysis, software design, programming, testing, and maintenance. MBTI has also been part of the research done by DaCunha and Greathead[18], Greathead[19].

Relying on one of the most widely used models of personality, Buchanan[15] explores the impact of the Big Five personality patterns on group cohesiveness and group performance on creative tasks and establishes patterns of three Big Five traits (Extraversion, Openness to Experience and Conscientiousness) as potential predictors of group performance on creative tasks. Kanij et al. [20] based their work on the question of "whether the personality of software testers may be different to other people involved in software development?" and to test this hypothesis they collected personality profiles using the Big Five factor model of a large group of software testers and a large group of people involved in other roles of software development. Their results indicate that software testers present a significantly higher conscientiousness factor than other software development practitioners.

Although neither MBTI[1] nor the Big Five are considered by all psychologists to be universally accepted [21], many researchers are employing them for a variety of purposes [22].

Studies such as those conducted by Yarkoni[10], Golbeck et al.[11] and Gill[12] have sought to identify personality traits from text (blogs, twitter, email). As referred by Gill[12], "Personality is projected linguistically" and "Personality can be perceived through language". The way people write and speak and the words they use relate to their personality traits, so one can say there is a strong relationship between personality and the use of language, especially when people write or talk about topics of their choice[10].

3. METHODOLOGY

3.1 Datasets

Building on the work done by Gonzalez-Barahona et al.[6] we used the data of the Eclipse project[2] available at [3], with information from the following repositories: source code management (git), issue tracking (Bugzilla), mailing lists (archived in mbox format), and code review (Gerrit). From the dumps that are provided by Metrics Grimoire, the databases were restored and the datasets used in the experimental stage were built.

Since we are interested in identifying relationships between social and technical aspects in the evolution of FLOSS projects, the source code repository and the mailing lists are the most relevant data for the purpose of this work. Specifically, we used the data of the Eclipse Platform subproject, which in turn is divided into the following components [7]: Ant - Eclipse/Ant integration, Workspace (Team, CVS, Compare, Resources) - Platform resource management, Debug - Generic execution debug framework, Releng - Release Engineering, Search - Integrated search facility, SWT - Standard Widget Toolkit, Text - Text editor framework and UI - Platform user interface, runtime and help components.

3.2 Socio-Technical Analysis Methodology

Because of the specificity of the study we conducted, it was necessary to define a methodology to study socio-technical relationships in FLOSS projects. The methodology we propose starts by defining the best representation of the data describing the social and technical aspects of the developers in the software development process to, thereafter, build the datasets to be used in the experimental stage. The representation we used for technical data was binary vectors. Each vector represents whether a committer touched each file of the project or not. For personality data, the repre-

[1] en.wikipedia.org/wiki/Myers-Briggs_Type_Indicator#Criticism

[2] www.eclipse.org/eclipse

[3] gsyc.es/~jgb/repro/2015-msr-grimoire-data

sentation was the personality traits characterizing software developers, which they project through their emails. An exploratory analysis was performed to become familiar with the data and to identify potential inconsistencies that should be corrected.

For each research question we wanted to answer, a specific experiment was configured and carried out. The first experiment was intended to answer RQ1: What personality traits can be identified through communications between software developers involved in FLOSS projects?, so that, at this stage, IBM Watson Personality Insights becomes more prominent. The dataset consisted of emails sent by committers to the mailing lists of the Eclipse Platform project and their subprojects.

The goal of the second experiment was to answer RQ2: What personality traits stand out according to the projects the software developers are involved in? and, at this stage, we used the personality traits identified in the above stage, and using clustering techniques (k-means and spectral clustering), we identified the personality traits characterizing each of the resulting clusters.

Finally, the third experiment was intended to answer RQ3: What relationships can be observed between the social activities (communication through the project mailing lists) of the committers and personality traits characterizing the groups they belong to? For this purpose we created a graph (a social network) representing e-mail communication among committers. Using the results obtained in the above stages, we determined the more distinctive personality traits of the nodes connected to the hubs in the graph.

The methodology we proposed is depicted in Figure 1. The main steps, which are described in detail in the following section, were as follows:

- Restoring databases from the dumps (source code repository and mailing lists).
- Datasets construction (social, technical and personality).
- Exploratory data analysis.
- Identifying technical and personality groups by applying clustering techniques.
- Identifying personality traits that characterize each of the technical groups.
- Visualization of social (communication) networks.
- Identification of social and technical relationships.

3.3 Tools

For clustering we used scikit-learn[4], for plotting we used matplotlib[5], for scientific computing we used NumPy[6] and SciPy[7]; for data manipulation we used pandas[8]; and for network visualization we used NetworkX[9].

On the other hand, with regard to the study of social aspects identified from communications among software developers, the tool we used was IBM Watson Personality Insights[10]. IBM Watson Personality Insights service [10, 23] [11] can detect personality traits reflected in text written by a subject. This was particularly useful for this work since it was unfeasible to apply a personality test to each of the committers who contribute to the FLOSS project under study.

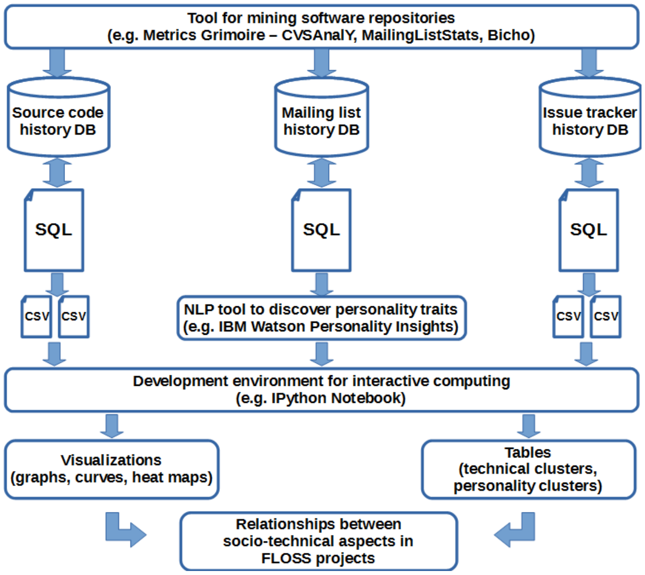
Tool for mining software repositories
(e.g. Metrics Grimoire – CVSAnalY, MailingListStats, Bicho)
Source code history DB
Mailing list history DB
Issue tracker history DB
SQL
SQL
SQL
CSV
CSV
NLP tool to discover personality traits
(e.g. IBM Watson Personality Insights)
CSV
CSV
Development environment for interactive computing
(e.g. IPython Notebook)
Visualizations
(graphs, curves, heat maps)
Tables
(technical clusters, personality clusters)
Relationships between socio-technical aspects in FLOSS projects

Figure 1: Socio-Technical analysis methodology diagram.

4. EXPERIMENTS AND RESULTS

4.1 Exploratory data analysis

To learn about the data to be used in the experiments, we conducted an exploratory analysis summarized in Table 1. The date range for which data were obtained is between January 1st, 2003 and January 1st, 2015.

No	Project / Subproject	Committers	Commits	Mailing List Senders	Mailing List Messages
1	Eclipse Platform	46	6829	405	939
2	Platform Text	33	5911	71	454
3	Platform UI	112	25110	375	5069
4	RelEng	4	205	232	22716
5	Resources	28	3077	180	1561
6	SWT	46	21984	1125	5967

Table 1: Eclipse Platform project - Number of registers

[4] scikit-learn.org

[5] matplotlib.org

[6] www.numpy.org

[7] www.scipy.org

[8] pandas.pydata.org

[9] networkx.github.io

[10] www.ibm.com/smarterplanet/us/en/ibmwatson/developercloud/personality-insights.html

[11] www.ibm.com/smarterplanet/us/en/ibmwatson/developercloud/doc/personality-insights/science.shtml

4.2 Technical and personality groups

To find technical and personality groups of data objects that share similar characteristics, we conducted cluster analysis through spectral clustering. The algorithm receives as a parameter the number of clusters (k) in order to partition a dataset. We look for this parameter through the elbow curve by plotting the result of the within-cluster sum of squared errors (SSE) for different values of k.

Looking at the point at which the SSE value changes significantly, we selected $k_t = 5$ for technical clustering and $k_p = 3$ for personality clustering. Just to clarify, a technical clustering corresponds to the result of applying the clustering algorithm to the data representing the files a committer has touched (i.e. a file modified by a committer and sent by him to the repository). Personality clustering refers to the result of applying the clustering algorithm to the data representing personality traits inferred from committers' texts (emails sent by committers to mailing lists).

The data representation (binary vectors representing if a committer touched or not a file of a project) suggests that it is more convenient to use a similarity metric like Jaccard similarity coefficient (used in this work) than Euclidean distance.

Tables 2 and 3 show the results for technical clustering. The projects touched by committers in technical clusters were obtained averaging the number of times that project directories have been touched by the committers in each cluster. Only values that represent a participation or contribution of the committers to the project greater than or equal to 7% are taken into account.

Technical cluster	Number of committers
0	107
1	11
2	12
3	24
4	13
Total	168

Table 2: Results of technical clustering.

Technical cluster	Eclipse Platform	Eclipse Platform Runtime	Eclipse Platform SWT	Eclipse Platform Team	Eclipse Platform Text	Eclipse Platform UI
0	*	*	0.13	*	0.07	**0.67**
1	**0.24**	0.09	*	*	*	**0.56**
2	*	*	**0.73**	*	*	0.19
3	0.07	0.07	0.08	**0.27**	0.12	**0.39**
4	*	*	*	*	*	**0.84**

* *Values < 0.07*

Table 3: Results for technical clustering. Averaged number of times that project directories have been touched by committers.

In addition, we calculate, for each technical cluster, the number of committers who touched each project, as shown in Table 4. Hence, to understand the meaning of technical groups we consider the results presented in Tables 3 and 4. We noticed that most committers from all technical clusters, except for technical cluster 2, contribute to Eclipse Platform UI project. In fact, there is great participation of technical cluster 0 (83 committers) and a high activity of the technical cluster 4 (0.84) with reference to this project. Analyzing participation in other projects, we observed that committers from cluster 0 tend to be more present in the Eclipse Platform SWT project (35) just as committers from cluster 2 (12), while committers from cluster 1 lean toward Eclipse Platform Runtime (10), and committers from cluster 4 tend to work in Eclipse Platform Team project (24). Furthermore, we noted uniformity in cluster 4 as all the committers belonging to this group (13) contribute to Eclipse Platform SWT, Eclipse Platform Team, Eclipse Platform Text and Eclipse Platform UI projects, with more activity in the latter project (0.84).

Technical cluster	Eclipse Platform	Eclipse Platform Runtime	Eclipse Platform SWT	Eclipse Platform Team	Eclipse Platform Text	Eclipse Platform UI
0	11	17	**35**	22	19	**83**
1	7	**10**	6	5	7	**11**
2	2	0	**12**	7	11	**12**
3	13	12	13	**24**	14	**24**
4	9	6	**13**	**13**	**13**	**13**

Table 4: Number of committers touching projects in technical clusters.

Table 5 shows the results for personality clustering, and the heat map in Figure 2 depicts the results of each Big Five dimension and facet, each Need, and each Value (rows) by each personality cluster (columns). Because of space restrictions, we show just the top-10 (the lowest) entropy values for Big Five dimensions and facets, needs, and values. As recommended by the IBM Watson Personality Insights service and for statistically significant results, we analyzed at least 3,500 words written by each committer. To get enough text for each committer, we concatenated his/her e-mails sent to the project mailing lists.

Personality cluster	Number of committers
0	42
1	24
2	2
Total	68

Table 5: Results of personality clustering.

Then, as answer to RQ1, the personality traits can be identified through communications between software developers involved in FLOSS projects, which are those corresponding to the Big Five dimensions and facets, needs, and values[12]. As highlighted in the heat map, personality cluster 2 groups the committers with the highest scores in personality traits such as Extraversion (92%), Orderliness (91%), Trust (89%), Cautiousness (70%) and Dutifulness (68%),

[12] www.ibm.com/smarterplanet/us/en/ibmwatson/developercloud/doc/personality-insights/models.shtml

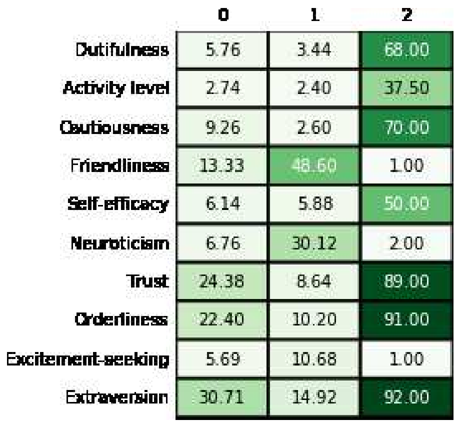

Figure 2: Heat map of the most discriminative factors for the personality clustering.

and the lowest values in Excitement-seeking (1%), Friendliness (1%) and Neuroticism (2%). Personality traits characterizing cluster 1 by its moderately high values are Friendliness (48.6%) and Neuroticism (30.12%), opposed to cluster 3 which has very low values in those personality dimensions. Finally, personality traits standing out in cluster 0 are Trust (24,38%), Orderliness (22.40%) and Extraversion (30.71%), which are lower than those of cluster 2, but higher than those of cluster 1.

4.3 Personality traits characterizing technical groups

Each personality cluster has associated personality traits that characterize and distinguish its members. From the results of the IBM Watson Personality Insights service it is possible to identify which personality traits are dominant in each cluster, becoming differentiating features, and what personality traits have similar values across all groups. In addition, we know which technical group is associated to each committer.

By computing the entropy for each of the Big Five dimensions and facets, Need and Values, it is possible to determine which of these attributes provide more information or become a differentiating factor when analyzing the technical groups, depending on the personality traits of the committers who are part of them. The lower the entropy, the greater the variation of the values of the corresponding attribute for the technical clusters, i.e., the attribute turns out more informative. This allows us to characterize the group or groups in which it is presented, and in which we must focus on when making an analysis of each cluster.

Since we know the technical cluster where each committer belongs and the personality traits for committers belonging to each technical cluster, we can compute the centroids of personality traits for each technical cluster. Again, due to space restriction, we show just the 10 lowest values and the 10 highest values of entropy for the Big Five dimensions and facets, Needs, and Values of personality centroids computed by averaging the values of the personality traits of committers in each technical cluster. Figure 3 shows the results.

From the results reported in Figure 3 and Table 3, we can answer RQ2. Personality traits scoring high ($\geqslant 80\%$) and with nearly uniform values through all technical clusters (e.g. *Cooperation*, *Sympathy*, *Conscientiousness*, *Achievement striving*, *Cautiousness*, *Openness*, *Adventurousness*, *Imagination*, *Intellect*, *Liberalism*, *Conservation* and *Self-enhancement*) could be considered as personality factors characterizing the project, i.e. people involved in the project will most likely exhibit high values in these personality traits. On the other hand, personality traits scoring lower ($< 25\%$) allow us to identify relationships with the technical aspects, differentiating personality features among the different technical clusters.

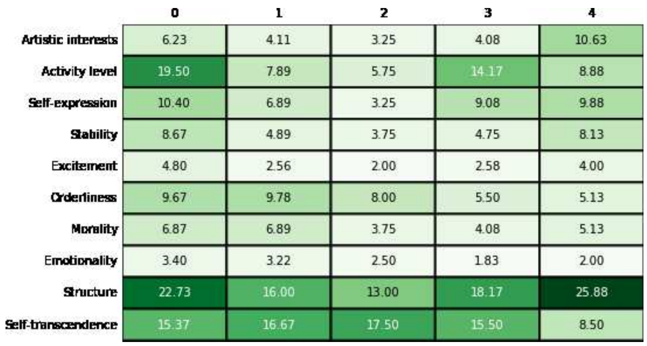

Figure 3: Heat map of personality centroids for each technical cluster.

Figure 4 summarizes personality traits by technical cluster allowing to visualize which personality traits are more representative in each technical cluster. From this representation, one can notice the dominant facets for the different technical clusters. For instance, committers grouped in the technical cluster 4 score high values in the *Artistic interests* facet in comparison with other clusters, and they mainly contribute to a project related to graphical elements, i.e., Eclipse Platform UI. Furthermore, a high value in *Structure* need[13] (25.88%), and a low value in *Self-transcendence* value[14] (8.5%) regarding the other clusters could explain why the committers of the technical cluster 4 contribute to only one project.

4.4 Visualizing the social network - from committers to mailing lists

Using the e-mails sent by committers to the Eclipse Platform project mailing lists, we built a graph representing e-mail communications. The graph in Figure 5 shows committers and mailing lists (PlatformDev, Search, Text, Core, Releng, UI, SWT, Team, i.e., red circles) as nodes. The thickness of the edge between a committer and a mailing list represents the amount of emails sent by the committer to the list. Additionally, the color of the nodes representing committers corresponds to the personality cluster to which the committer belongs to. Only committers that have sent more than 10 e-mails to any of the lists were taken into account.

Figure 5 helps us to answer RQ3. Committers belonging to personality group 0 (purple circles) are those distributed through all mailing lists, except Team. This may be attributed to the ranking they have in traits such as Altruism (73.14%), Cheerfulness (70.93%), Gregariousness

[13] https://www.ibm.com/smarterplanet/us/en/ibmwatson/developercloud/doc/personality-insights/models.shtml#outputNeeds

[14] https://www.ibm.com/smarterplanet/us/en/ibmwatson/developercloud/doc/personality-insights/models.shtml#outputValues

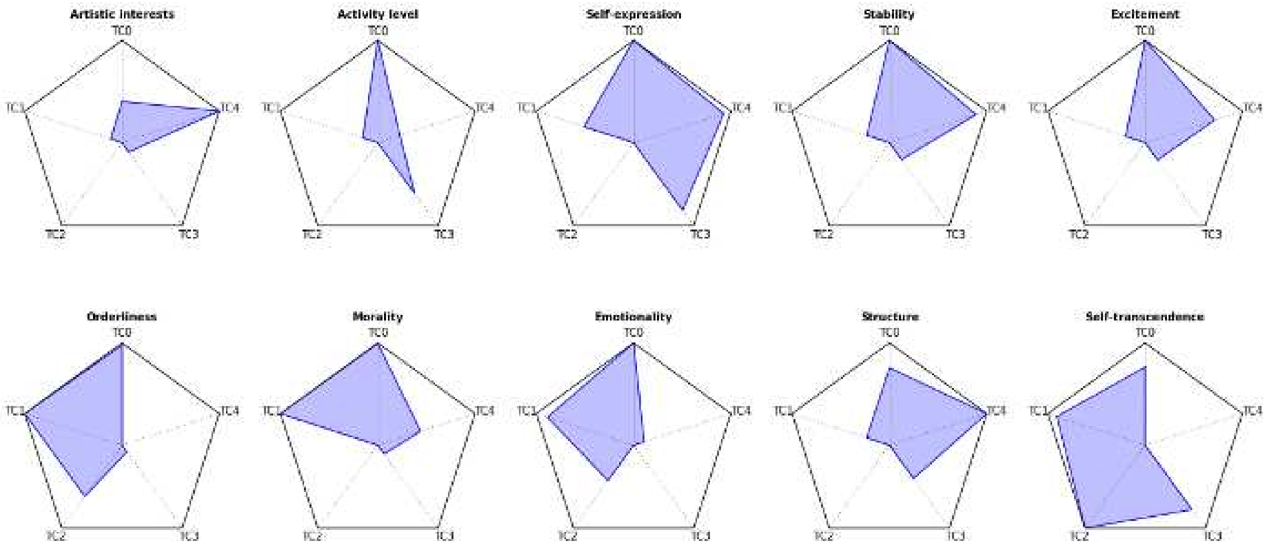

Figure 4: Radar charts of personality traits by technical cluster (TC).

(89.05%), and Self-discipline (46.62%). The top 5 of committers, sorted by the number of messages sent to mailing lists, is distributed between representatives of the personality clusters 0 and 1. This tendency to actively participate in the lists may be related to the traits having similar values in those clusters such as *Gregariousness* (89.05% for cluster 0 and 76.88% for cluster 1), and *Altruism* (73.14% for cluster 0 and 63.04% for cluster 1). Only one committer appears in the graph of Figure 5 representing the personality group 2, which has the highest value of *Cautiousness* (9.26% for cluster 0, 2.60% for cluster 1, and 70% for cluster 2) and the lowest value of *Cautiousness* (1%), compared to the other groups (13.33% for cluster 0 and 48.60 for cluster 1); but surprisingly this group has the highest value of *Extraversion* (30.71% for cluster 0, 14.92% for cluster 1, and 92% for cluster 2) and for *Dutifulness* (5.76% for cluster 0, 3.44% for cluster 1, and 68% for cluster 2).

5. CONCLUSIONS

FLOSS projects are characterized by a high component of social interaction where a large number of people with great technical skills contribute from different parts of the world, in most cases without knowing each other. Within this context we conducted a preliminary study aimed at uncovering the factors involved in the formation of working groups and the dynamics of communication that occur during the process of software development.

Considering that personality traits influence most, if not all, of the human activities, we took this feature as the centerpiece of the work done. In this regard, services such as IBM Watson Personality Insights are crucial to analyze personality traits from text, when is impractical to apply personality tests to each participant of a study. By having the personality characteristics (a total of 52) inferred by the service, we were able to identify relationships established, either solely from personality traits as is the case of the groups presented in Table 5, or those established from both personality traits and social activities of the committers related to communication through the project mailing lists, as shown in Figure 5.

As evidenced by analyzing the graph representing social activities (Figure 5), is not enough to focus on just one personality trait to identify patterns due to the complexity of the personality and its constitutive factors. Thus, it is necessary to give a comprehensive and detailed look at each one of the dimensions, facets and categories of the three personality models (Big Five, Needs and Values) to be able to draw conclusions most closely related to the behavior the data try to show us.

6. THREATS TO VALIDITY

What we must have in mind is that the main aim of this work is to explore whether it is possible to extract personality traits from developer e-mails, and try to uncover relationships among those traits and the social and technical activities performed by the software team. As a feasible way to achieve this goal, we proposed a novel approach to collect, process, and analyze the relevant data, which involves the use of several tools and clustering techniques.

We are aware that our preliminary results may be affected by several validity threats inherent in the proposed approach. To mention only the most important ones, our results depend on an automatic analysis of developer e-mails performed by IBM Watson Personality Insights service, instead of personality assessment questionnaires designed by psychologists and applied directly to the software team members. Moreover, our experiment is limited to the mailing lists and code base of one system only. Thus, variables such as the project domain, the system size, the team size, and the quality and availability of the text could influence the effectiveness of our approach.

7. REFERENCES

[1] F. Mairesse, M. A. Walker, M. R. Mehl, and R. K. Moore, “Using Linguistic Cues for the Automatic Recognition of Personality in Conversation and Text,” J. Artif. Intell. Res., vol. 30, no. 1, pp. 457 – 500, 2007.

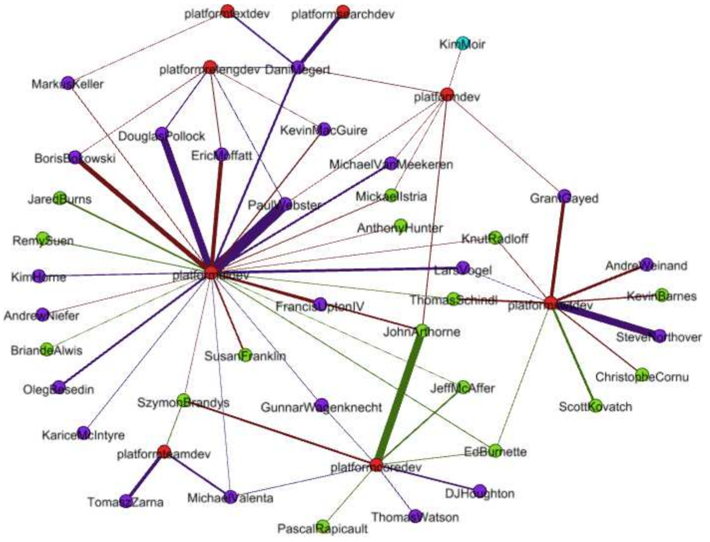

Figure 5: Social (email communication) network. From committers to mailing lists.

[2] S. Sheppard and B. Curtis, “First-year results from a research program on human factors in software engineering,” Proc. Natl. Comput. Conf., pp. 1021–1027, 1979.

[3] V. R. Basili and R. W. Reiter, “An Investigation of Human Factors in Software Development,” Computer (Long. Beach. Calif)., vol. 12, no. 12, pp. 21–38, Dec. 1979.

[4] S. Wagner and M. Ruhe, “A systematic review of productivity factors in software development,” Language (Baltim)., 1980.

[5] I. Sommerville and T. Rodden, “Human, social and organisational influences on the software process,” Software Process, 1996.

[6] J. M. Gonzalez-Barahona, G. Robles, and D. Izquierdo-Cortazar, “The MetricsGrimoire Database Collection,” in 2015 IEEE/ACM 12th Working Conference on Mining Software Repositories, 2015, pp. 478–481.

[7] Eclipse Project. URL: https://eclipse.org/eclipse/

[8] Fernando Pérez, Brian E. Granger, IPython: A System for Interactive Scientific Computing, Computing in Science and Engineering, vol. 9, no. 3, pp. 21-29, May/June 2007, doi:10.1109/MCSE.2007.53. URL: http://ipython.org

[9] IBM Watson Personality Insights service. URL: www.ibm.com/smarterplanet/us/en/ibmwatson/developercloud/personality-insights.html

[10] T. Yarkoni, “Personality in 100,000 Words: A large-scale analysis of personality and word use among bloggers,” J. Res. Pers., vol. 44, no. 3, pp. 363–373, Jun. 2010.

[11] J. Golbeck, C. Robles, M. Edmondson, and K. Turner, “Predicting Personality from Twitter,” in 2011 IEEE Third Int’l Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third Int’l Conference on Social Computing, 2011, pp. 149–156.

[12] A. J. Gill, “Personality and Language: The projection and perception of personality in computer-mediated communication,” University of Edinburgh, 2003.

[13] S. T. Acuña and N. Juristo, “Assigning people to roles in software projects,” Softw. Pract. Exp., vol. 34, no. 7, pp. 675–696, Jun. 2004.

[14] J. H. Bradley and F. J. Hebert, “The effect of personality type on team performance,” J. Manag. Dev., vol. 16, no. 5, pp. 337–353, Jul. 1997.

[15] L. Buchanan, The impact of big five personality characteristics on group cohesion and creative task performance. PhD dissertation, Virginia Polytechnic Institute and State University. 1998. Retrieved from https://vtechworks.lib.vt.edu/bitstream/handle/10919/30415/etd.pdf?sequence=1&isAllowed=y.

[16] L. F. Capretz and F. Ahmed, “Making Sense of Software Development and Personality Types,” IT Prof., vol. 12, no. 1, pp. 6–13, Jan. 2010.

[17] L. F. Capretz, “Personality types in software engineering,” Int. J. Hum. Comput. Stud., vol. 58, no. 2, pp. 207–214, Feb. 2003.

[18] A. Da Cunha and D. Greathead, “Code review and personality: is performance linked to MBTI type?,” Tech. Rep. Ser. . . . , p. 18, 2004.

[19] D. Greathead, “MBTI personality type and student code comprehension skill,” Proc. 20th Work. Psychol., p. 13, 2008.

[20] T. Kanij, R. Merkel, and J. Grundy, “An Empirical Investigation of Personality Traits of Software Testers,” in 2015 IEEE/ACM 8th International Workshop on Cooperative and Human Aspects of Software Engineering, 2015, pp. 1–7.

[21] L. Capretz, “Are software engineers really engineers?,” World Trans. Eng. Technol., 2002.

[22] S. Cruz, F. Q. B. da Silva, and L. F. Capretz, “Forty years of research on personality in software engineering: A mapping study,” Comput. Human Behav., vol. 46, pp. 94–113, May 2015.

[23] Yang H, Li Y. Identifying user needs from social media. IBM Research Division, San Jose. 2013.