PhD in Information Technology and Electrical Engineering
Università degli Studi di Napoli Federico II

PhD Student: Francesco Altiero

Cycle: XXXVI

Training and Research Activities Report

Year: Second

Tutor: prof. Adriano Peron

*__tutor signature_*

Co-Tutor: prof. Anna Corazza

Date: October 30, 2022

Training and Research Activities Report

PhD in Information Technology and Electrical Engineering

Cycle: XXXVI **Author: Francesco Altiero**

1. Information:

- **PhD student:** Francesco Altiero
- **DR number:** 995043
- **Date of birth: 07/07/1986**
- **Master Science degree:** Computer Science **University:** Federico II
- **Doctoral Cycle:** XXXVI
- **Scholarship type:** *UNINA*
- **Tutor:** prof. Adriano Peron
- **Co-tutor: prof.** Anna Corazza

2. Study and training activities:

Activity	Type[1]	Hours	Credits	Dates	Organizer	Cert.[2]
Supporting code-related tasks via Deep-Learning	Seminar	1	0.2	11.11.2021	Nargiz Humbatova, Software Institute - Università della Svizzera Italiana	N
Possible Quantum Machine Learning Approaches in HEP	Seminar	2	0.4	12.11.2021	Prof. A. S. Cacciapuoti, DIETI - Unina	Y
Connecting the dots: Investigating an ATP campaign using SPLUNK	Seminar	2	0.4	26.11.2021	Prof. D. Cotroneo, Prof. S. P. Romano, Dr. R. Natella, DIETI UNINA	N
Threat Hunting Use-Cases	Seminar	2	0.4	13.12.2021	Prof. D. Cotroneo, Prof. S. P. Romano, Dr. R. Natella, DIETI UNINA	N
All roads lead to WebRTC: an introduction to Janus	Seminar	2	0.4	16.12.2021	Prof. Simon Pietro Romano, DIETI UNINA	N
Picariello Lectures: Can a text-to-speech Engine Generate Human Sentiments?	Seminar	1	0.2	28.02.2022	Prof. Giuseppe Longo, DIETI UNINA	N
IEEE Authorship and Open Access Symposium: Tips and Best Practices to Get Published from IEEE Editors	Seminar	1.5	0.3	30.03.2022	Institute of Electrical and Electronics Engineers (IEEE)	Y

Picariello Lectures: Towards a Political Philosophy of AI	Seminar	1.5	0.3	11.04.2022	Giuseppe Longo, DIETI UNINA	N
An Introduction to Deep Learning for Natural Language Processing	Seminar	1	0.2	13.04.2022	Prof. Francesco Cutugno, DIETI UNINA	N
Explainable Natural Language Inference	Seminar	1.5	0.3	13.04.2022	Prof. Francesco Cutugno, DIETI UNINA	N
5G Networks in Action - The Private Mobile Era	Seminar	1.5	0.3	11.05.2022	Prof. Antonia Maria Tulino, DIETI	N
Fixed Wireless Access: Site Engineering, Implementation and Legal Regulation	Seminar	5	1	17.05.2022	Prof. Antonia Maria Tulino, DIETI	N
Vine robots: design challenges and unique opportunities	Seminar	1	0.2	31.05.2022	Dr. Mario Selvaggio, DIETI UNINA	N
Quantum computing with superconducting qubits, an overview on the current state and future directions at Rigetti computing	Seminar	1	0.2	20.06.2022	Prof. Francesco Tafuri, Dip. Di Fisica E. Pancini - UNINA	N
Privacy-Preserving Machine Learning	Seminar	2	0.4	14.10.2022	Prof. Simon Pietro Romano and Prof. Roberto Natella, DIETI UNINA	Y
Imprenditorialità Accademica	Course	27	4	26.05/13-14-20.06/13-26.07.2022	Prof. Pierluigi Rippa, DIETI UNINA	Y
Neural Networks and Deep Learning	Course	65	10	12-18-19-26.01/02-07-08-15-16-23.02/01-08-15.03.2022	Prof. Giorgio Carlo Buttazzo, Scola Superiore Sant'Anna, PISA	Y

1) Courses, Seminar, Doctoral School, Research, Tutorship
2) Choose: Y or N

2.1. Study and training activities - credits earned

	Courses	Seminars	Research	Tutorship	Total
Bimonth 1	**0**	**1.8**	**7**	**0**	**8.8**
Bimonth 2	**0**	**0.2**	**10**	**0**	**10.2**
Bimonth 3	**0**	**1.3**	**8**	**0**	**9.3**
Bimonth 4	**0**	**1.7**	**7**	**0**	**8.7**
Bimonth 5	**4**	**0**	**5**	**0**	**9**
Bimonth 6	**10**	**0.4**	**7**	**0**	**14.4**
Total	**14**	**5.4**	**44**	**0**	**63.4**
Expected	**10 -20**	**5 - 10**	**30 - 45**	**0 – 1.6**	

3. Research activity:

My main research topic is the *Regression Test Prioritization* (RTP) problem. The problem consists into re-ordering the test cases in a test suite upon the release of a newer software version, to discover the presence of *regression faults*, i.e., faults in pre-existing modules which were previously working.

I am currently working on the problem with a hybrid approach involving both *test coverage* and *code churn*: in a first step, the changes in the source code between these two versions are analyzed in order to produce a *quantitative* estimation of structural modifications, then the test suite is reordered scheduling test cases which covers more significative changes to be executed earlier. To evaluate the amount of structural changes between two versions of the source code, I chose to employ *Tree Kernel functions*, a machine learning technique which have been widely used in different fields of computer science (e.g., *Natural Language Processing* and *Software Engineering*). Tree Kernels can evaluate the structural similarity between two tree-based structure, and I applied them on *Abstract Syntax Trees* of the source code, a natural structured representation of blocks and statements of the source code.

Continuing with the research guidelines defined in the past year, I analyzed several scientifical paper on the RTP topic to have a more detailed insight on the current state-of-the-art of the approaches proposed in literature. As the field is highly empirical, techniques for RTP are often evaluated on collections of software projects and common metrics are used throughout literatures. Due to the fact that software projects with real faults are hard to find on publicly available code sharing platforms (e.g., because developers do not update the codebase until all test cases pass on their local machine), experiments performed in literature are somehow *in vitro*: some versions of chosen benchmark software are injected with artificial faults in order to measure the fault-exposing rate of re-ordered test suite. Only a small number of papers in literature are *in vivo*, i.e., propose experiments on projects with real faults. This latter kind of experiments can be more reliable, as are demonstrations of a real application of RTP techniques. Said that, I decided to perform experiments on datasets of both kind on the RTP techniques I am researching. This led to the creation of two datasets, one containing projects with injected faults and another one mined from software repositories and with real-world faults.

The dataset which I initially created in my first year as a Ph.D. student fall under the *in vitro* category. In this year, I extended this dataset to include a greater number of projects. As not all software projects can be suitable for the study, I managed to retrieve the datasets defined in two state-of-the-art papers which used injected faults. The retrieval of these projects was not a trivial process, as the source code of some of them were removed from their original location (e.g., no more available on *GitHub*) or there was no mean to re-execute their building phase to obtain additional information, such as coverage reports. Once several projects and different software versions for each project have been collected, I injected faults in the changed parts between two versions through Major, a Java mutation framework. Then, for each version, I partitioned the mutants generated by the framework in groups of 5 and defined 100 version variants. Each variant contained a group of faults and its source code has been modified to include such modification in the code.

I also managed to work on the definition of a dataset with real faults. I initially analyzed the dataset commonly used for RTP experiments with real faults, focusing in particular on *RTPTorrent* and *IRTorrent*. The analyzed datasets were however not suited for my studies: often they did not provide the original sources or some projects were not available. Thus, to progress in this research line, I needed to obtain benchmark projects with real faults through mining. To achieve this result, I developed, with my research team, a tool used to mine *GitHub* and *TravisCI*, an online CI/CD platform, in order to download the source code of software versions including at least a fault. The mining tool receives in input a list of projects and versions which can be found on both platforms, then analyzes their *TravisCI* building history to assess if at least one test case has failed during the execution of that particular build. If so, downloads the very version on GitHub along with its previous version. Then, it executes the building and test phase, producing also test coverage reports. We started the mining tool from the lists of software projects found in two RTP real-faulted datasets mentioned above, *RTPTorrent* and *IRTorrent*, collecting a dataset with 228 software versions for 22 Java programs. The dataset has been also published online at https://doi.org/10.5281/zenodo.5913165 and presented at the Mining Software Repositories 2022 Conference, to which I submitted the technical paper describing it.

Concerning the RTP techniques I am researching on, I analyzed some meta-heuristic frameworks and their application to the problem at hand. Specifically, I focused on Genetic Algorithms which are widely used in meta-heuristic approaches to RTP. I developed a novel genetic RTP technique, providing an original target function which measures the rate of changed covered lines (namely APTC on difference) and a novel crossover operator which breeds two permutation of a test-suite giving higher priority to those test cases which covers at least an element in the churn. The obtained results were promising, and I submitted a paper describing our study to the Euromicro SEAA22 conference, which was accepted.

Furthermore, I worked on the enhancing of my Tree Kernel RTP technique. In particular, I switched the granularity of the technique from statement level to method level, significatively reducing the execution time of the technique with marginal loss in rate of fault discovering. Furthermore, we defined a post-prioritization set, namely the Quotient-Set Prioritization, which re-orders test-cases using the score assigned by the TK Prioritization as an equivalence relation. The studies we performed on the in vitro dataset with injected faults showed more stability and generally better performances with respect to the TK Prioritization and compared to all our baseline techniques. For this reason, I am currently working with my team to describe the studies in a paper.

4. Research products:

Conference Paper: F. Altiero, A. Corazza, S. Di Martino, A. Peron and L. L. L. Starace, "ReCover: a Curated Dataset for Regression Testing Research," *2022 IEEE/ACM 19th International Conference on Mining Software Repositories (MSR)*, 2022. Published in proceedings, pp. 196-200, doi: 10.1145/3524842.3528490.

Conference Paper: F. Altiero, G. Colella, A. Corazza, S. Di Martino, A. Peron and L. L. L. Starace, "Change-Aware Regression Test Prioritization using Genetic Algorithms," *48th Euromicro Conference Series on Software Engineering and Advanced Applications (SEAA22)*, 2022. Accepted and currently in publication.

Dataset: *ReCover*: a software repository containing 228 software versions for 22 Java projects including full source code, coverage reports and with real faults to perform *in vivo* regression testing experiments. The dataset has been mined from *TravisTorrent* and *GitHub* using an *ad hoc* designed mining tool written in Java. Doi: 10.5281/zenodo.5913165.

Software: *Extension of Prioritization Platform*: a Java application which allows the execution of Regression Test Prioritization experiments. The platform has been extended to include *ReCover* as a dataset, along with definition of other state-of-the-art techniques used for comparison with novel approaches I'm currently researching.

5. Conferences and seminars attended

Workshop: *Ital-IA 2022 – Convegno Nazionale CINI sull'Intelligenza Artificiale* (ITAL-IA22), Turin, Italy, 10.02.2022. Attended online. Presented the paper *Fine-grained Source Code Similarity with Tree Kernels to Support Software Testing*.

Conference: *19th International Conference on Mining Software Repositories* (MSR 2022), Pittsburgh, PA, USA, 18-20.05.2022. Attended online. Presented the paper *ReCover: a Cured Dataset for Regression Testing Research* on 18.05.2022.

Conference: *48th Euromicro Conference Series on Software Engineering and Advanced Applications* (SEAA22), Maspalomas, Gran Canaria, Spain, 31.08-2.09.2022. Attended in presence. Presented the paper *Change-Aware Regression Test Prioritization using Genetic Algorithms* on 2.09.2022.

6. Activity abroad:

None. 0 months spent abroad.

7. Tutorship

None.

8. Plan for year three

For my third Ph.D. year I plan to extend my studies in Regression Test Prioritization. With a dataset of software projects with real faults, I want to execute the prioritization techniques I am currently developing on real case scenarios. This kind of experimentation will give me more insights on these techniques and will let me cope with issues which can possibly arise in real-world cases. To increase the generalizability of my findings, I also plan to execute the mining tool again, in order to retrieve a larger number of real-world projects on which experiment.

Furthermore, I plan to extends Tree Kernel based prioritization techniques by taking into account also the *semantic* of changes, rather than structural changes only. This can be done by particular Tree Kernel functions, namely *Smoothed Partial Tree Kernel*, which can be tuned to highlight more critical changes in the source code (e.g., a change in the control flow should be weighted more than a simple variable renaming). To realize the tuning process, I need a big enough amount of data, which I can obtain by mining software repositories.

I am also interested into an abroad research period, to collaborate with other researchers in my field and to share information and thoughts. Two destinations I have in mind are the University College of Dublin and the Technischen Universität Wien, in which there are heterogeneous research teams currently studying the regression testing problem.

To what concerns tutorship activities, in agree with my tutor, I plan to attend some practical lessons for the course of *Algorithms and Data Structures*, a 2nd year course related to the bachelor's degree in computer science at DIETI. The lessons should be focused on practical problems on the topic and to ease the students' efforts of learning concepts which will be central for their future career.

For my Ph.D. thesis, I want to present my research on Tree Kernels applied to Regression Test Prioritization. Based on the experience of both the first and the second year and to what I plan to achieve in the third year, I will discuss the fundaments beneath Tree Kernel based prioritization strategies, the methodology behind the experimental setting, including the need to retrieve a significative dataset, and the results of my research in the application of these techniques compared with the baselines.