Investigating Samples Representativeness for an Online Experiment in Java Code Search

Rafael Maiani de Mello
Federal University of Rio de Janeiro
P.O. Box 68511, Brazil
+55 21 3938 8712
rmaiani@cos.ufrj.br

Kathryn T. Stolee
Iowa State University
209 Atanasoff Hall
Ames, IA 50011
kstolee@iastate.edu

Guilherme Horta Travassos
Federal University of Rio de Janeiro
P.O. Box 68511, Brazil
+55 21 3938 8712
ght@cos.ufrj.br

***Abstract*— *Context:* The results of large-scale studies in software engineering can be significantly impacted by samples' representativeness. Diverse population sources can be used to support sampling for such studies. *Goal:* To compare two samples, one from the crowdsourcing platform Mechanical Turk and another from the professional social network LinkedIn, in an online experiment for evaluating the relevance of Java code snippets to programming tasks. *Method:* To compare the samples (subjects' experience, programming habits) and experimental results concerned with three experimental trials. *Results:* LinkedIn's subjects present significantly higher levels of experience in Java programming and programming in general than Mechanical Turk's subjects. The experimental results revealed a significant difference between samples and suggested that LinkedIn's subjects were more pessimistic than Mechanical Turk's subjects despite a high level consistency in the experimental results. *Conclusion:* The combined use of sources of sampling can bring benefits to large scale studies in software engineering, especially when heterogeneity is desired in the population. Thus, it can be useful to investigate and characterize alternative sources of sampling for performing large-scale studies in software engineering.**

Keywords*— *experimental software engineering; sampling; population; survey; sampling frame

I. INTRODUCTION

In statistics, a *sampling frame* is the source from which a *sample*, i.e. a subset of units from a study *population* can be retrieved [1]. In the context of Software Engineering (SE) research, primary studies are often conducted over samples established by convenience [2,3,4]. Student classes, research groups and organizational units are common sampling frames from which individuals have been recruited to collaborate in SE *quasi*-experiments. As a consequence, the external validity of the evidence observed in such studies is significantly limited.

Although the specialized nature of some SE problems allows them to be investigated through qualitative strategies such as action research [5] and case studies [6], there are many open research questions that could be better answered through large-scale experiments and surveys, in which the *representativeness* of the sample can significantly impact the results. Unlike areas in which the units of observation are controlled and can be applied in diverse experimental arrangements, SE research is hampered by a lack of available sampling frames composed by representative populations of individuals or groups of individuals, such as organizations and project teams [4, 7]. One can see that not only the variability of SE research contexts contributes to this scenario [8], but also the business context of SE practice.

In this context, online experiments represent a good opportunity to investigate alternative *sources of sampling* [9] from which better adequate sampling frames can be established to support specific research contexts. A possible immediate contribution expected on using such sources is related with the increase of samples' size, but it is not limited to. It is also expected that *representative samples* should be sufficiently *heterogeneous* from the point of view of the *attributes* previously established to characterize each individual from a specific study population [10].

Two first trials from an experiment on evaluating Java code snippets from three distinct search engines (*Google*, a source-code specific search engine, *Merobase*, and a research prototype, *Satsy*) were conducted [11, 12]. Although the operationalization and the protocol of such trials presented some differences, both used as population the anonymous workers from the crowdsourcing platform Amazon's Mechanical Turk (MTurk).

Then, a third trial was conducted having as population the members from a group of interest from the professional social network LinkedIn (www.linkedin.com). This trial applied a systematic plan to recruit a random and geographically distributed sample from such group, following concepts from a framework originally developed to support researchers on establishing representative samples in large scale SE surveys [9].

This work presents this third trial and examines the contributions on using LinkedIn as source of sampling in comparison with samples and results obtained in the previous trials using MTurk. The contributions of this work are:

1. Operational replication of a study on code search results using LinkedIn for sampling (previously performed using MTurk).

978-1-4673-7899-4/15/$31.00 ©2015 IEEE

2. Comparison of subjects' samples obtained for Java code search studies using both platforms.
3. Comparison of experimental results obtained for Java code search studies using both platforms.
4. Recommendations based on evidence for software engineering researchers on the use of these sources of sampling in their studies.

The rest of the paper is organized as follows: Section 2 presents the related work and background in source of sampling, MTurk and LinkedIn. Section 3 characterizes the studies, describing how samples were composed and how the subjects were recruited in each trial. Section 4 presents the comparison between LinkedIn and MTurk samples while Section 5 presents the comparison between LinkedIn and MTurk results. Section 6 presents a discussion about the results presented in the two previous sections. Section 7 discusses the threats to validity of the presented investigation and Section 8 presents the conclusions.

II. BACKGROUND AND RELATED WORK

This section presents background information on the source of sampling concept as well as information on the platforms used for the studies.

A. Source of Sampling

Investigating alternatives to suppress the lack of sampling frames for supporting SE surveys, de Mello, et al. proposed the concept of *source of sampling*, a database (automated or not) from which adequate *subpopulations* of the *target population* can be *systematically retrieved* and *randomly sampled* [9]. Each source of sampling should have at least one type of *search unit*, i.e., the unit from which one or more *units of observation* can be retrieved from it. The following four Essential Requirements (ER) shall be satisfied to a source of sampling *candidate* be considered *valid*:

- *ER1. A source of sampling shall not intentionally represent a segregated subset from the target population*, i.e., for a target population "X", it is not adequate to search for units from a source *intentionally designed* to compose a specific subset of "X".
- *ER2. A source of sampling shall not present any bias on including on its database preferentially only subsets from the target population.* Unequal criteria for including search units means unequal sampling opportunities.
- *ER3. All sources of sampling search units and their units of observation must be unique and identifiable.*
- *ER4. All sources of sampling search units must be accessible. If there are hidden search units, it is not possible to contextualize the population.*

There are also nine desirable requirements (DR), three concerned with the source accuracy (ADR), two concerned with its clearness (CDR) and four regarding its completeness (CoDR). The description of DRs and the other concepts from a conceptual framework for supporting sampling in SE surveys can be found at [9].

Although the concept of source of sampling was originally designed to the context of survey research, one can see that it is not limited to it, since large-scale experiments also need to deliver representative samples. The following subsections briefly introduce the platforms used as sources of sampling in the studies and discuss their feasibility to be applied considering the mentioned requirements. It also briefly explores how SE research has previously been supported by such platforms.

B. Mechanical Turk

Mechanical Turk (https://www.mturk.com) is commonly characterized as a *crowdsourcing* platform from which paid tasks can be performed by registered collaborators (workers). The main goal of MTurk is to provide a safe and simple environment in which online workers can earn money by performing HITs (Human Interaction Tasks). The payment rules (including each task value) are previously established by the *requester*, i.e. the individual who created the task. The MTurk environment also allows requesters to manage the HIT in many ways, which includes applying a previous qualification task prior to participation and manually validating each completed task before payment.

1) Sampling and Worker Characteristics

As previously evaluated by de Mello et al. [9], one can see that the essential requirement *ER4* can't be supported by MTurk since the workers are anonymous to the requesters and they cannot be previously characterized as part of a population. Thus, the recruitment process in MTurk is blind (indirect recruitment), similar to posting recruitments "on the street walls". *ER2* also can't be supported since the acceptance of new workers is restricted to decision of MTurk team on accepting each subscription [9]. In fact, the minimum requirements to be accepted as an MTurk worker are not clear. For instance, one of the Brazilian authors tried to subscribe on MTurk in 2013 and again in 2014. In both cases, his subscription was not accepted by MTurk. However, a third request in 2015 was accepted.

A 2010 investigation regarding the characterization of MTurk workers indicated that they shifted from a primarily moderate-income, U.S.-based workforce towards an increasingly international group with a significant population of young and well-educated Indian workers [13]. However, it was observed that many tasks from diverse social fields applying MTurk are commonly restricted to U.S. citizens. In the context of political science research, Berinsky et al. [14] calls MTurk as a source of "convenience samples", assessing through a series of comparative studies the potential advantages and limitations of using such platform as U.S. citizens' recruitment source. The authors observed that relative to other convenience samples often used in the area for U.S. citizens, MTurk subjects are often more representative of the general population and substantially less expensive to recruit. However, many limitations were also reported,

such as that MTurk samples are younger and present significantly different representative population.

2) Mechanical Turk and SE Research

Crowdsourcing in software engineering is typically used to perform a task rather than evaluate research. For example, MTurk alone has been used for research in program synthesis [15], GUI testing [16], program verification [17], and fault localization [18]. For evaluating software engineering research, MTurk has been used to assess the readability and other properties of software patches [19], express preferences between refactored and smelly web mashups [20], create input/output specifications for code search in SQL and web mashups [21], and evaluate the relevance of code snippets to programming tasks [11, 12].

Although demographics investigations of MTurk samples in the SE context have not been conducted, Layman et al. [22] pointed out eight recommendations on using MTurk for SE user studies in order to deal with the following potential threats to validity: *Qualifications of subjects*, *Data validation, Procedure adherence, and Independence of observations.* To date, research evaluating SE research has taken two approaches to control response quality, using a qualification exam (e.g., [11, 12, 20, 21]) or dropping responses that fall outside of one standard deviation of the mean (e.g., [19]). One study compared the performance of students in a classroom to participants on MTurk performing the same tasks in SQL or Yahoo! Pipes [21]. Overall, no differences were observed in the results between the samples. Considering just the Yahoo! Pipes tasks, however, there was a difference in accuracy between students and the MTurk population. As no comparative study was performed regarding the samples characteristics, it is unclear if the differences in results are due to differences in the study platform (online vs. in a classroom), sample characteristics, or some other factor.

C. LinkedIn

LinkedIn (http://www.lnkedin.com) is currently considered the world's largest professional social network in the world, having more than 250 million members. LinkedIn has been actively used for supporting headhunting activities, connecting co-workers and classmates, disseminating job opportunities, and hosting forums regarding many areas of knowledge. Any professional can subscribe to LinkedIn and each one is responsible for maintaining his/her own profile updated, in a way as a dynamic and interactive *curriculum vitae*. For instance, the connections from a LinkedIn user can give endorsements regarding user skills and make a recommendation about the user.

1) Sampling and Member Characteristics

Having memberships in groups of interest is a common practice between LinkedIn users. Such groups can be created by a user to support one or more purposes such as the following, identified through the descriptions from a large set of SE groups [10, 23]: promoting the discussion between practitioners regarding a specific technology or technique (worldwide or by region), promoting a specific organization and connecting its collaborators, jobs offering, promoting events such as congresses and fairs. Membership requests to large-scale groups of interest are typically automatically accepted but it can be changed by the group owner. As a group of interest member, a user can interact with the other group members participating from its discussion topics and keeping in touch with a specific member through individual messages.

In spite of LinkedIn broad coverage, especially in the context of SE community, it presents several restrictions on filtering and accessing members, even when using "premium" account plans (the basic subscription is free). First of all, our experience showed that it was not possible to filter more than few hundred members from a specific group of interest or from the whole network. In addition, a user probably is not able to keep in touch with a LinkedIn member without sharing something in common with him/her, such as a group or a connection.

Thus, considering the feasibility of using LinkedIn as a source of sampling, two types of search unit may be considered: *individuals* and *groups of interest*. In this context, due to the restrictions already mentioned, it was observed that LinkedIn can't be used as a valid source of sampling using individuals as search units since ER4 can't be supported [9]. On the other hand, there are no restriction on searching groups of interest and accessing their data. In addition, as already mentioned, since the user is a member from a group of interest, it is possible to keep in touch with other group' members. Thus, it was concluded that LinkedIn can be used as a source of sampling when *group of interest* is the search unit [9]. However, it is important to emphasize that some additional efforts on filtering and recruiting subjects may be required due to some LinkedIn technical restrictions, as exemplified by de Mello et al. [10, 23]

2) Linkedin and SE research

One survey was found in SE literature using LinkedIn in the sampling process to post generic invitation messages in group forums without establishing a controlled sampling frame [24]. In the second trial from a survey on Requirements Effort Estimation, de Mello and Travassos [25] initially used the forum posting strategy, but after observing the limitations of sampling control in such strategy, the authors decided to establish a random sample composed by 996 subjects from two also randomly selected groups of interest. Such groups were obtained from a sampling frame composed by groups of interest identified through a systematic sampling plan. As a result, it was observed evidence that the LinkedIn sample presented a more heterogeneous profile and similar experience level than another sample obtained by convenience.

Based on the lessons learned in the previous study, a more detailed sampling plan was designed and applied to

support sampling for a third trial of a Survey on Characteristics of agility in Software Processes. In this trial 7,745 subjects were recruited from 19 LinkedIn groups of interest distributed into 8 strata [10, 23]. In total, 291 subjects answered the survey and higher heterogeneity of LinkedIn samples in comparison with the samples obtained in the two first survey executions was observed, when only paper authors were invited [26]. Such heterogeneity was essential to reinforce some results observed in the previous trials and put another ones in doubt [27]. The lessons learned in both surveys mentioned in this subsection supported the researchers on proposing the already mentioned framework [9].

III. The Experiment Trials

In an effort to evaluate the relevance of code search results for a new code search algorithm, called *Satsy*, Stolee and Elbaum designed an experiment[1] to evaluate the relevance of source code snippets to various programming tasks. The first implementation of this study, called MT1, used MTurk and the results of three search algorithms were compared [12]. This study was then replicated using a sample from LinkedIn, called LI1, presented in this work.

While a qualification exam was used for MT1, the characteristics of the samples were not obtained. Thus, a second study on MTurk, MT2 was designed and executed on a different version of Satsy, comparing just two search algorithms [11]. The same qualification exams were used for MT1 and MT2, except MT2 obtained characteristics of the sample. This section briefly characterizes these trials following the classification presented by Gómez et al. [28]. The main differences between the experimental dimensions of such trials are discussed.

A. MT1

In the first trial of the experiment, three search queries were issued to each of three search algorithms for each of eight programming tasks. The search algorithms and their respective query formats were:

1. *Google: keyword queries*
2. *Merobase: method signature queries*
3. *Satsy: input/output examples as queries*

For each query to each search engine, the top 10 source code results were obtained. This resulted in 720 code snippets for evaluation (i.e., 3 search algorithm * 3 queries * 8 tasks * 10 results). In the MTurk environment, 30 HITs were created, each containing 24 source code snippets. Each HIT presented a programming task followed by three code snippets, one from each search approach. It required the participant to state whether the snippet is relevant to the task, and why, and whether the snippet solves the task, and why. An example of a snippet and the relevance questions are in Figure 1. The participant was not made aware of which snippet came from which search. The order of the programming tasks, was randomized across HITs.

[1]https://sites.google.com/site/semanticcodesearch/publications/generalizing-ranking.

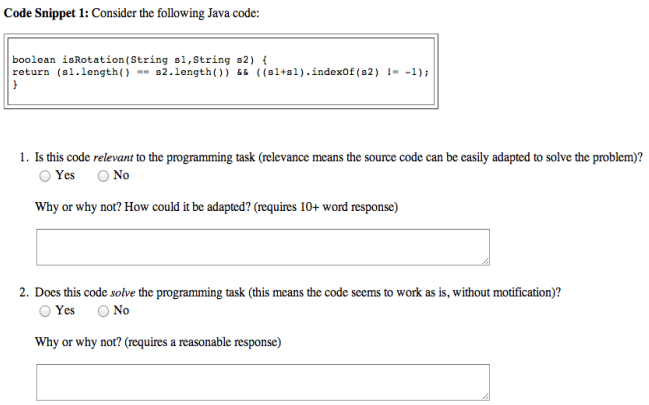


Fig. 1. Example Task and Code Snippet in MTurk

Prior to participation, workers had to complete and pass a *qualification exam*, in which were asked four Java competency questions (of which at least two needed to be answered correctly to pass). If passed, the subjects were delivered the informed consent. MT1 participants were paid $3.25 for each HIT completed, with a maximum of one HIT per participant (8 programming tasks).

B. MT2

MT2 also used MTurk as source of sampling but presented a different *operationalization* since it was compared only the results from Satsy to results from Google, excluding Merobase [11]. MT2 also present a different *protocol* since a different set of programming tasks (*experimental objects*) were applied and MT2 included a subject characterization questionnaire that should be answered before performing the experimental task (*instruments*).

In addition to indirect recruitment, both MT1 and MT2 studies were advertised using a post on the HITsWorthTurkingFor page of reddit.com. Although each HIT from MT2 contained only one programming task, all 64 HITs were available for all participants. For each completed task, $0.50 were paid. In both MT1 and MT2 payment was only given if the tasks performed were manually verified as satisfactory.

C. LI1

LI1 followed the same operationalization from MT1 but varying on the population since the professional social network LinkedIn was used as source of sampling. LI1 also presents a different *protocol* from the previous studies, without using the "why" questions and including the same subject characterization questionnaire and experimental objects applied in MT1. Since MTurk environment is not always accessible worldwide (as is needed for sampling), a new environment was created that presented participants with the same code snippets and questions from MT1 but removing the "why" from each task.

In fact, for MTurk studies, the "why" questions were put in place to prevent participants from haphazardly answering yes/no on the questions. Further, in a pilot of LI1, from 100 LinkedIn members recruited, no one

participant completed the study, presumably because it took too long and the compensation was not sufficient. Since the LinkedIn participants were directly recruited and not performing the tasks for payment, we made the assumption that fewer controls were needed to ensure the quality of results. Thus, the "why" questions were removed for LI1.

LinkedIn subjects were invited individually to perform each of the eight tasks in 20 minutes or less through individual messages using the LinkedIn environment. Although no direct reward was offered for the subjects, it was advertised that a donation of $1.00 for Brazilian Red Cross would be performed for each subject that completed all eight tasks. Each task had one programming task each, as compared to MT1 where a single HIT had eight programming tasks. The order of the eight programming tasks in LI1 was identical to the order of programming tasks in MT1 HITs.

1) Sampling plan

LI1 followed a detailed plan applying the already mentioned conceptual framework [9]. While LinkedIn was established as the source of sampling, its *groups of interest* were defined as search units. Then, the biggest group (in number of members) identified in LinkedIn as devoted to Java programming (*Java Developers*) was selected to be the sampling frame.

At the time of the recruitment, 182,288 professionals working with *Information Technology, Computer Software* and *Telecommunications* composed this sampling frame. Then, it was observed that approximately 90% of such professionals (165,134) were from 40 distinct countries, as presented in Table I. A sample was composed based in the distribution of members from such countries by each geographic region, as presented in Table I. To calculate the presented sample sizes, a Confidence Level of 95% and a Confidence Interval of 6 points were considered, being applied the following formulas (1) and (2) for calculating the sample size considering correction for finite population [30]:

$$SS_i = \frac{Z^2 \times p \times (1-p)}{c^2} \quad (1) \qquad SS_f = \frac{SS_i}{1+\frac{SS_i - 1}{p}} \quad (2)$$

where Z = (Z-value for 95% of confidence level), p=0.5 (percentage picking a choice, expressed as decimal; 0.5 used for sample size needed) and c=0.06 (confidence interval). Due to an operational error on calculating the sample size, a total of 380 members were randomly sampled from the European countries (instead of 265). Thus, a total of 1,657 individuals were invited through individual messages sent from LinkedIn.

D. Comparing the Trials' Plans

Having the original experiment as baseline, Table II shows the main differences between MT1 and MT2/LI1 regarding the four dimensions of study plans and their elements presented by Gómez et al. [28]. Following the classification proposed by the authors MT2 can be classified as a *changed-operationalization/protocol replication* from MT1 while LI1 can be classified as a *changed-population/protocol/experimenters replication* from MT1.

TABLE I. DISTRIBUTION OF LI1 SAMPLE BY GEOGRAPHICAL REGIONS.

Region	Countries	Number of Members	Sample Size
Asia	India, Pakistan, China, Israel, Philippines, Indonesia, Bangladesh	60,605	266
USA+ Canada	USA and Canada	51,757	265
Europe	United Kingdom, Italy, Turkey, France, Ukraine, Spain, Poland, Romania, Netherlands, Portugal, Russian Federation, Ireland, Belgium, Sweden, Czech Republic, Switzerland, Bulgaria, Serbia, Greece, Hungary, Denmark, Finland and Norway	38,684	265
Latin America	Brazil, Argentina and Mexico	8,427	259
Africa	South Africa, Egypt, Morocco	3,248	247
Oceania	Australia	2,413	240
Sum		**165,134**	**1,542**

TABLE II. COMPARISON BETWEEN THE CHARACTERIZATION OF THE DIMENSIONS/ ELEMENTS FROM THE THREE TRIALS.

Dimension	Element	MT1 X MT2	MT1 X LI1
Operationalization	Cause	≠	=
	Effect	=	=
Population	Subjects properties	=	≠
	Objects properties	=	=
Protocol	Design	=	=
	Experimental objects	≠	=
	Guides	=	≠
	Instruments	=	≠
	Data Analysis Techniques	=	=
Experimenters	Designer, Trainer, Monitor, Measurer, Analyst	=	≠

One can observe through Table II that results from MT1 could not be aggregated/compared with results from MT2 since its operationalization were different. On the other hand, as mentioned in subsection III.B, the instruments of MT1 did not included the subjects' characterization. Thus, in order to support the aimed comparison between MTurk and LinkedIn samples/ results, the comparisons highlighted in Figure 2 were performed.

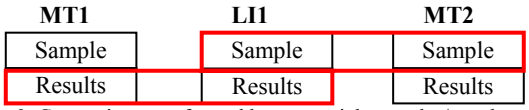


Fig. 2. Comparisons performed between trials samples/ results.

Table III synthetizes the main characteristics of the recruitment strategies applied in each trial, including in which extent persuasive factors were applied [29]. Such characteristics should take into account on interpreting the comparisons presented in the next subsections. While MT1

and MT2 followed similar recruitment strategies, LI1 was more rigorous on giving some reward and imposing a more restricted time limit. Although only LI1 subjects were personally invited, they were informed about the execution date limit (scarcity) and about the relevance of their participation for strengthening the study results. Thus, since similar persuasive factors were applied in MT1 and MT2 and the qualification test applied in all trials was the same, our expectation is that the MT1 and MT2 participant characteristics are similar. In fact, based on MTurk workers IDs, it was found that that 30% of workers from MT1 also participated in MT2 and are thus represented in the comparison of samples.

TABLE III. CHARACTERIZATION OF THE RECRUITMENT STRATEGY USED IN EACH TRIAL.

Factor	MT1	MT2	LI1
Personal Invitation	No	No	Yes
Scarcity	No	No	Yes
Identification of the researchers	Yes	Yes	Yes
Reward	$3.25	$0.50	$1.00
Type of reward	Payment	Payment	Donation
Tasks per participant	1	64	8
Minimum of tasks for reward	1	1	8
Time Limit to complete all tasks	60 minutes	None	20 minutes

IV. SAMPLE EVALUATION

This section describes a comparison study performed between the effectiveness of the candidates from MTurk and LinkedIn in the qualification exam and the characteristics of the effective samples obtained through each source. As explained in subsection III.D, MTurk will be represented by MT2 candidates and its sample in this comparison. Regarding the qualification exam results, the following hypotheses emerged:

- *H01*. There is no association between the source of the potential subjects and the qualification exam results
- *HA1*. There is association between the source of the potential subjects and the qualification exam results

Figure 3 shows the qualification exam questions analyzed for this paper. Four of the five questions are open-ended and the remaining question, Q3, was a single-select multiple choice question. The qualification exam results will be compared based on the distributions of the approving and rejecting for each sample. For such comparison, it will be applied the Pearson's chi-square test (alpha=0.01).

Related to the comparison between the samples from MT2 and LI1 (respectively SMT2 and SLI1), the following hypotheses emerged:

- *H02*. There is no difference between the experience level of SMT2 and SLI1
- *HA2*. SMT2 experience level is different from SLI1
- *H03*. There is no difference between the programming habits of SMT2 and SLI1
- *HA3*. SMT2 has different programming habits from SLI1

Q1: *How many years of programming experience do you have?*
Q2: *How many years of Java programming experience do you have?*
Q3: *How often do you program?*
daily
weekly
monthly
never
Q4: *How many search results do you typically examine before finding something useful?*
Q5: *How many different search queries do you try before finding a useful result?*

Fig. 3. Qualification Exam Questions.

The samples characteristics will be observed in this experiment in terms of the following attributes:

- *Years of experience on programming (Q1)*
- *Years of experience programming in Java (Q2).*
- *Frequency of programming (Q3, scale: daily, weekly, monthly, never)*
- *Amount search results typically examined before finding something useful (Q4)*
- *Amount of different search queries tried before finding a useful result (Q5)*

Since normality were not observed in some of the distributions analyzed, it was decided to apply the non-parametric *Mann-Whitney Wilcox test of means* for supporting all samples evaluations with $\alpha = 0.01$.

MT2 had 84 subjects interested in the experiment (candidates), answering the qualification exam. From these, only 19 effectively participated in the study, composing SMT2. LI1 had 114 subjects interested in the experiment and 83 effectively participated, performing at least one task. Then, such participants compose SLI1.

A. Qualification Exam

Table IV shows the distribution of candidates from MT2 and LI1 qualified and not qualified to the experiment. It was a simple and fast qualification exam, having four basic questions about java programming. If a candidate answered two or more Java questions correctly, he/she was considered "qualified" to perform the experimental tasks.

TABLE IV. QUALIFICATION EXAM RESULTS BY SOURCE OF THE CANDIDATE.

Study	Qualified	Not Qualified	Total
MT2	75	9	84
LI1	112	3	115

One can see that less than 3% of the candidates from LI1 were not approved in the exam, while more than 10% of the subjects from MT2 were also not approved. Applying the Pearson's' chi-square test over these distributions it was observed that the performance in the qualification exam is associated to the source of the candidate, with a p-value of 0.018. Thus, it was possible to refute *H01* and accept *HA1*, regarding the association

between the source of the potential subjects and the qualification exam results.

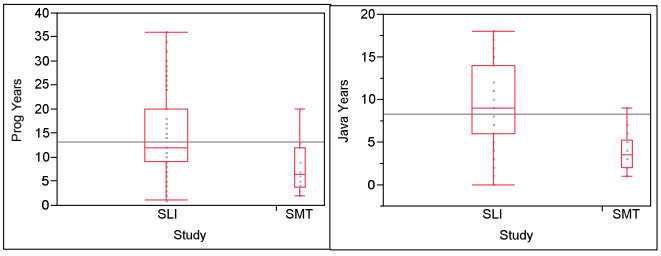
B. Experience Level

After removing three outliers, the ranges, medians and means for the distributions of *programming experience* were calculated, as presented in Table V. The numbers in parenthesis indicate the total number of outliers removed in each distribution. Boxplots in Figure 4(a) presents the distribution of programming experience in both samples.

One can see that the range of years of programming in SMT2 is contained in the larger range of years from SLI1, suggesting that the distribution in SMT2 is more diverse. The results from the Mann-Whitney test indicates that SLI1 has significantly higher distribution of programming experience than SMT2 with p-value= 0.0004.

TABLE V. PROGRAMMING EXPERIENCE BY SAMPLE

Sample	Size	Mean	Std. Dev.	Median	Min.	Max.
SMT2	18(1)	7.39	4.79	6.50	2	20
SLI1	81(2)	14.62	8.41	12	1	36



(a) (b)

Fig. 4. Programming experience and Java Experience by sample.

Regarding *Java programming experience*, just one outlier needed to be removed. As can be observed in Table VI, the distribution of ranges suggests that SLI1 is more diverse than SMT2. At the same time, one can see through the boxplots presented in Figure 4(b) how SMT2 is concentrated in the range of 1-4 years of Java programming experience. The results from the Mann-Whitney test indicates that SLI1 has significantly higher distribution of Java programming experience than SMT2 with p-value< 0.0001.

TABLE VI. JAVA PROGRAMMING EXPERIENCE BY SAMPLE

Sample	Size	Mean	Std. Dev.	Median	Min.	Max.
SMT2	18(1)	4.06	2.46	3.5	1	9
SLI1	83(0)	9.28	4.99	9	0	18

Thus, considering the results observed for experience level, it was possible to reject *H02* and accept *HA2* (experience level).

C. Programming Habits

Table VII shows the distributions of *frequency of programming* by the scale used in the questionnaire. One can see that most of respondents in SMT2 and SLI1 have the habit of programming daily (68% and 70%, respectively). Since *Weekly (SMT2)*, *Monthly (SMT2)* and *Never (SMT2 and SLI1)* distributions present insufficient sizes to apply the chi-Square test, it was decided to combine such values in a single value (Not Daily), reducing the degrees of freedom of the analysis for 2. As a result, it wasn't found evidence regarding the influence of the samples in the frequency of programming reported by the subjects.

TABLE VII. DISTRIBUTIONS OF FREQUENCY OF PROGRAMMING BY SAMPLE

Frequency	SMT2	SLI1
Daily	13	58
Weekly	3	17
Monthly	2	5
Never	1	1

Regarding the question *"How many search results do you examine…"*, a significant number of outliers was removed (19) considering the sample sizes. Then, it was observed that SMT2 presents less diverse behavior than SLI1 (Table VIII and Figure 5). The results from the Mann-Whitney test indicates that SLI1 has significantly different distribution of number of searches from SMT2 with p-value= 0.0017.

TABLE VIII. NUMBER OF SEARCH RESULTS BY SAMPLE.

Sample	Size	Mean	Std. Dev.	Median	Min.	Max.
SMT2	14(5)	2.429	0.64	2.5	1	3
SLI1	69(14)	3.696	1.53	3	0	8

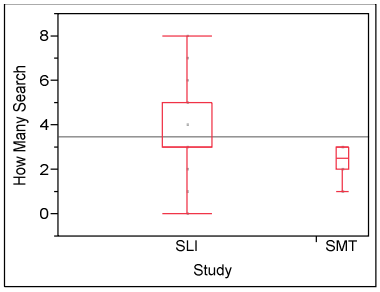


Fig. 5. Number of searches by sample.

Regarding the *question "How many search queries do you try..."*, 16 outliers were removed from the samples. By Table IX, one can see that SMT2 and SLI1 present similar ranges and closely means. Applying the Mann Whitney test, no significant difference between the distributions was found (p-value=0.1168).

TABLE IX. NUMBER OF SEARCH QUERIES BY SAMPLE.

Sample	Size	Mean	Std. Dev.	Median	Min.	Max.
SMT2	16(3)	2.188	0.65	2	1	3
SLI1	70(13)	2.543	0.86	3	1	4

Thus, considering the results observed only to the number of search results, it was possible to reject *H03* and accept *HA3* (programming habits).

V. RESULTS EVALUATION

This section describes a comparison study performed between the results obtained from MTurk and LinkedIn

samples. As explained in subsection III.D, MTurk will be represented by MT1 in this comparison, emerging the following hypotheses:

- *H04:* There is no difference between the experimental results from MT1 and LI1
- *HA4:* The results from MT1 and LI1 differ

Prior work used the relevance/precision of the top 10 results from a search, or P@10, as the metric for evaluating the relevance of search results [12]. We carry that forward in this study, comparing the P@10 for each combination of search approach, query, and programming task. In total, this creates 72 values for each study, the original and the replicated.

We measure the quality of search results using the relevance of the top-10 (P@10), where a participant in each trial (MT1 or LI1) determined the relevance for each of the top 10 search results, given a particular programming task. For each trial, we compute 72 values, one for each combination of search approach, query, and programming task.

As shown in TableX, the average P@10 value across MT1 is higher than that from LI1. Additionally, using the Shapiro-Wilk test of normality, with MT1, we reject the null hypothesis that the data are normally distributed, whereas with LI1, we do not reject the null hypothesis. For this reason, in comparing the results, we use non-parametric tests.

Considering the aggregated P@10 results from both studies, we use the *Mann-Whitney Wilcox test of means*, and find that H04 is rejected with $\alpha = 0.01$ ($p = 0.002$). This indicates that the experimental results from the two studies, MT1 and LI1, are significantly different.

We also use a 3-factor ANOVA with P@10 as the dependent variable, considering factor A (sample), factor B (algorithm) and factor C (programming task). The sample factor and algorithm factor are significant at $\alpha = 0.001$, indicating that the differences in P@10 values are not likely due to chance, but rather due to the sample and algorithm. The programming task is significant at $\alpha = 0.05$, as is the interaction between algorithm and programming task. Based on the F-ratio for the sample factor, this provides further evidence to reject *H04* and accept *HA4*.

While the difference between samples is significant, we also look at the general trends in the experimental results between MT1 and LI1. Between samples and within each search algorithm, we compared the relevance of results. Table X presents the average P@10 for each sample and search algorithm, as well as a hypothesis test evaluating the equality of sample means within the same search approach. For example, given Satsy, H0: $\mu MT1 = \mu LI1$ is not rejected at $\alpha = 0.01$ with p-value = 0.0533. For both samples, we see that the P@10 for Google > Satsy > Merobase. This consistency in results across the samples provides higher confidence in the results of the original study. When comparing the samples within a search algorithm, there is a significant difference between the Google results with $\alpha = 0.01$. For the other search algorithms, the differences are not significant.

TABLE X. AVERAGE P@10 ACROSS ALL PROGRAMMING TASKS AND QUERIES. HYPOTHESIS TEST USES MANN-WHITNEY WILCOX

Search Algorithm	Mean(P@10)		H0 test (p-value)
	MT1	LT1	
Google	0.675	0.519	0.0045
Satsy	0.533	0.390	0.0533
Merobase	0.375	0.305	0.3002

Within a sample and comparing the means between search algorithms, MT1 showed a significant difference between Google and Merobase with $\alpha = 0.01$ using the Mann-Whitney test, as shown in Table XI. At the same significance level, there was also a difference between Google and Merobase within LI1. In MT1, there were significant differences between Google and Satsy and between Satsy and Merobase at $\alpha = 0.05$. While there was a significant difference observed between Satsy and Google in LI1 at $\alpha = 0.05$, there was no observed difference between Satsy and Merobase for LI1. These trends seem relatively consistent among the samples, with a difference appearing only when comparing Satsy and Merobase in each sample.

Overall, despite high-level significant statistical differences between the samples regarding P@10 values, few differences are found when evaluating the experimental results within a sample, beyond the actual values. Even though the P@10 for LI1 was typically lower than P@10 for MT1, the search algorithm and sample were found to be significant factors in the ANOVA, Google and Merobase were shown to be significantly different, and Google > Satsy > Merobase.

TABLE XI. TESTS OF MEANS BETWEEN SEARCH ALGORITHMS WITHIN MT1 AND LI1 USING MANN-WHITNEY WILCOX TEST (**ALPHA = 0.01, * ALPHA = 0.05)

MT1 x LI1	Google	Satsy	Merobase
Google	-	0.0252*	0.0005**
Satsy	0.0457*	-	0.1767
Merobase	0.0001**	0.0326*	-

VI. DISCUSSION

The analyzed distributions of experience suggest that LinkedIn allowed us to retrieve a more diverse sample, considering the ranges of programming experience. This supports HA1, indicating there is association between the source of the potential subjects and the qualification exam results. One can see that the sample size retrieved and the choice of random sampling from the most represented interest group in Java was helpful to reach this result. In fact, considering the goal of MTurk to provide an environment from which online workers can earn money by performing HITs and observed payment values for HITs in general, it makes us wonder about the possibility of lower senior professional participation when compared with Linkedin when used in the context of SE research.

The sample analysis showed that candidates from LinkedIn were more effective in the qualification exam

than MTurk candidates. However, programmers reporting few years of experience in both samples participated in the experiment. Such results indicate the relevant contribution of the qualification exam as a complementary platform to the characterization questionnaire, in order to mitigate the introduction of "noise" in the experimental results. On the one hand, a probable reason to find more rejections in MTurk exams is due to its characteristic of crowdsourcing platform, allowing the anonymous participation of any worker. Workers may answer the exam questions quickly just to see if they can get an easy qualification. On the other hand, the plan applied to find a representative population in LinkedIn, allows identifying professionals related with Java programming.

From H04, the differences in experimental results between the MTurk and LinkedIn samples and the significance of the sample factor in the ANOVA indicate a benefit to using multiple sampling techniques and replication in experimental design. While the general trend in averages was the same for both samples (Table X), there were statistically significant differences. However, the high-level results – that the general trend of Google -> Satsy -> Merobase – is consistent among platforms and samples, providing greater confidence in the results of prior studies [12]. Still, aggregating these data seems appropriate to increase the sample size and to generalize the results across multiple samples and platforms.

When it comes to empirical studies with human participants, larger studies can provide more data points that can be useful for the research. That said, the participant characteristics must be appropriate for the study. While several participants from MTurk and LinkedIn qualified for the study, we found that the LinkedIn participants had significantly more programming experience than the MTurk participants.

Yet, the MTurk platform allowed more flexibility in the type and quantity of questions asked. As stated in Section 3.2, the LinkedIn replication study omitted the "why" questions, which can provide insight to why a participant decided a code snippet was relevant or not. These qualitative answers can provide important feedback to guide the research [31]. For example, programmers often commented on naming conventions in explanations of whether a code snippet is relevant to the programming task. Even so, observing the same general trend in experimental results between the platforms does increase our confidence in the original experimental results [12].

VII. Threats to Validity

As internal threat to validity of this comparative study, we highlight the use of samples from operational replications (MT1 and MT2) to compare, respectively with sample of and the results obtained in LI1.

One can also see that removing the "Why" questions in LT1 and the different recruitment strategies in the experiments can influence the interest of the members from both platforms on participating. Thus, some of the observed differences in experimental results could be the result of different instruments rather than different samples. However, observing the clear higher diversity of profile in the sample from LT1, we can argue that the differences in the recruitment strategies were not deterministic on stimulating only subsets of the target population.

Regarding the experiments, since MTurk participation was anonymous, one can see that it was not possible to control if a LI1 subject also participated in MT1 or MT2. However, we understand that due to the overall population sizes in both platforms and the small number of participants, the probability of such risk is minimum.

The operational error reported on calculating the sample size from Europe in LI1, can be also considered a threat to validity. However we emphasized that it doesn't affected the randomness of the sampling process. In addition, due to the observed low participation rates in all regions, one can see that subjects' location was not used to grouping the study results.

VIII. Conclusions

In this paper, a comparison between the samples and results obtained for operational replications of an online large scale experiment on evaluating Java code snippets was presented. The experiment trials used different sources of sampling (MTurk and LinkedIn) aiming to enlarge the size and heterogeneity of participants. Two trials (an original and operational replication) were used to characterize results and samples obtained with MTurk. Another operational replication of the original experiment in Linkedin was used to compare its results (with original study) and sample (with MTurk operational replication). Table XII summarizes the main findings of this comparative study, considering the samples analyzed.

TABLE XII. Main Findings from Comparative studies Presented in this Paper.

Hypothesis	Tests	MTurk	LinkedIn
H01	Qualification Exam	-	+
H02	Programming Experience	-	+
	Java Programming Experience	-	+
H03	Frequency of programming	No difference	
	Searching code habits	Different	
H04	Experimental results	Different	
	P@10 Search algorithms	Google -> Satsy -> Merobase	Google -> Satsy -> Merobase
	Google P@10	Different	
	Satsy P@10	No difference	
	Merobase P@10	No difference	

The observed results indicate that these sources of sampling present interesting features that can support SE studies however they differ in their capacity on providing samples for such studies. For instance, MTurk indicated to support samples with less experienced/more optimistic

participants when compared with LinkedIn. Besides, MTurk does not allow the characterization of the sampling frame, what can impose risks for those experiment trials demanding more thoughtful and specialized participation. Heterogeneity is also different between these two sources of sampling, on which Linkedin presents higher possibility of getting more heterogeneous samples than MTurk. Linkedin allows random sampling while MTurk does not.

Although MTurk do not satisfy all source of sampling essential requirements, their use combined with LinkedIn (having groups of interest as search unit) can bring benefits when considering the design and performing of large scale studies in software engineering. However, the decision about using one or another should be based on the characteristics of the study, the need for more or less experienced participants, expected heterogeneity and sampling strategy. Thus, the obtained results suggest the relevance of investigating and characterizing alternative sources of sampling for performing software engineering large-scale studies.

ACKNOWLEDGMENT

We thank the anonymous reviewers for their helpful feedback and the technical support of Pedro Correa da Silva. This research was supported in part by CNPq (305929/2014-3), NSF SHF-EAGER-1446932, NSF SHF-1218265, and the Harpole-Pentair endowment at Iowa State University.

REFERENCES

[1] C. E. Sarndal, B. Swensson, and J. Wretman, Model Assisted Survey Sampling, 1st ed. Springer, 1992.

[2] D.I. Sjøberg et al., "A survey of controlled experiments in software engineering". IEEE Transactions on Software Engineering, vol. 31(9), pp.733-753, 2005.

[3] T. Dybå, V.B. Kampenes and D.I. Sjøberg, "A systematic review of statistical power in software engineering experiments". Information and Software Technology, vol. 48, pp. 745–755, 2010.

[4] R.M. de Mello and G.H. Travassos. "Characterizing Sampling Frames in Software Engineering Surveys," In: Proc. 12th Workshop on Experimental Software Engineering (ESELAW), 2015. Available at: http://eventos.spc.org.pe/cibse2015/pdfs/01_ESELAW15.pdf.

[5] P.S.M Santos and G.H. Travassos, G. H. "Action research use in software engineering: An initial survey," In: Proc. ESEM 2009, 10.1109/ESEM.2009.5316013, IEEE, 2009.

[6] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," Empirical Software Engineering vol. 14(2): pp. 131-164, 2009.

[7] R.M. de Mello and G.H. Travassos, "An ecological perspective towards the evolution of quantitative studies in software engineering," In: Proc. EASE 2013, doi: 10.1145/2460999.2461031 ACM, 2013.

[8] K. Petersen and C. Wohlin, "Context in industrial software engineering research," In: Proc. ESEM 2009. 10.1109/ESEM.2009.5316010, IEEE, 2009.

[9] R.M. de Mello, P.C. da Silva, P. Runeson and G.H. Travassos, "Towards a framework to support large scale sampling in software engineering surveys," In: Proc. ESEM 2014, pp. 48-52, doi: 10.1145/2652524.2652567 ACM, 2014.

[10] R.M. de Mello, P.C. da Silva and G.H. Travassos, "Investigating Probabilistic Sampling Approaches for Large-Scale Surveys in Software Engineering," Journal of Software Engineering Research and Development vol. 3:8, doi:10.1186/s40411-015-0023-0, 2015.

[11] K.T. Stolee, S. Elbaum and D. Dobos, "Solving the Search for Source Code," ACM Transactions on Software Engineering Methodology vol. 23(3) Art.26, 45 pp, doi: 10.1145/2581377, 2015.

[12] K.T. Stolee, S. Elbaum and M.B. Dwyer, "Code Search with Input/Output Queries: Generalizing, Ranking and Assessment," The Journal of Systems and Soft., doi:10.1016/j.jss.2015.04.081, 2015.

[13] J. Ross et al., "Who are the crowdworkers?: shifting demographics in mechanical turk," In: Proc. HFCS 2010, pp. 2863-2872, doi: 10.1145/1753846.1753873, ACM, 2010.

[14] A.J. Berinsky, G.A. Huber and G.S. Lenz, "Evaluating online labor markets for experimental research: Amazon.com's Mechanical Turk," Political Analysis, vol. 20.3 (2012), pp. 351-368.

[15] R. A. Cochran, L. D'Antoni, B. Livshits, D. Molnar, and M. Veanes, "Program boosting: Program synthesis via crowd-sourcing". In: Proc. ACM SIGPLAN-SIGACT SPPL, 2015, pp. 677–688.

[16] E. Dolstra, R. Vliegendhart, and J. Pouwelse, "Crowdsourcing GUI tests". In: Proc. ICSTVV 2013.. 2013, pp. 332–341.

[17] T. W. Schiller and M. D. Ernst, "Reducing the barriers to writing verified specifications". In: Proc. OOPSLA 2012, pp. 95–112L.

[18] Z. P. Fry and W. Weimer, "A human study of fault localization accuracy,". In: Proc. ICSM 2010, 2010.

[19] Z. P. Fry, B. Landau, and W. Weimer, "A human study of patch maintainability". In: Proc. ISSTA 2012, pp. 177–187.

[20] K.T. Stolee and S. Elbaum, "Exploring the use of crowdsourcing to support empirical studies in software engineering". In: Proc. ESEM 2010, doi: 10.1145/1852786.1852832, ACM, 2010.

[21] K.T. Stolee, and S. Elbaum. "On the use of input/output queries for code search. In: Proc". In: Proc. ESEM 2013, IEEE, 2013.

[22] Layman, L., G. Sigurdsson, "Using Amazon's Mechanical Turk for User Studies: Eight Things You Need to Know". In: Proc. ESEM 2013, 10.1109/ESEM.2013.42, IEEE, 2013.

[23] R.M. de Mello, P.C. da Silva and G.H Travassos, "Investigating Probabilistic Sampling Approaches for Large-Scale Surveys in Software Engineering". In: Proc. ESELAW 2014, 2014.

[24] M.E. Joorabchi, A. Mesba and P. Kruchten, "Real challenges in mobile app development". In: Proc. ESEM 2013, IEEE, 2013.

[25] R.M. de Mello and G.H. Travassos, "Would Sociable Software Engineers Observe Better?". In: Proc. ESEM 2013, doi: 10.1109/ESEM.2013.33, IEEE, 2013.

[26] R.M. de Mello, P.C. da Silva and G.H. Travassos, "Sampling improvement in software engineering surveys,". In: Proc. ESEM 2014, pp. 13-17, doi: 10.1145/2652524.2652566, ACM, 2014.

[27] R.M. de Mello, P.C. da Silva and G.H. Travassos, "Agilidade em Processos de Software: Evidências Sobre Características de Agilidade e Práticas Ágeis." In: XIII SBQS, SBC, Brazil, 2014 (in Portuguese)

[28] O.S, Gómez, N. Juristo and S. Vegas, "Understanding replication of experiments in software engineering: A classification," Information and Software Technology, vol. 56.8, pp. 1033-1048, doi:10.1016/j.infsof.2014.04.004, 2014.

[29] E. Smith et al. "Improving developer participation rates in surveys,". In: Proc. CHASE 2013, doi: 10.1109/CHASE.2013.6614738, IEEE. 2013.

[30] L.L. Kupper and K.B. Hafner, "How appropriate are popular sample size formulas?" The American Statistician, v. 43(2), 101-105, 1989.

[31] K.T. Stolee, J. Saylor and T. Lund, "Exploring the benefits of using redundant responses in crowdsourced evaluations". In: Proc. CSI-SE/ICSE 2015, doi: 10.1109/CSI-SE.2015.15, IEEE, 2015.