Master of Science in Software Engineering
March 2017

Investigating Metrics that are Good Predictors of Human Oracle Costs

An Experiment

Kartheek Arun Sai Ram Chilla
Kavya Chelluboina

Faculty of Computing
Blekinge Institute of Technology
SE–371 79 Karlskrona, Sweden

This thesis is submitted to the Faculty of Computing at Blekinge Institute of Technology in partial fulfillment of the requirements for the degree of Master of Science in Software Engineering. The thesis is equivalent to 20 weeks of full time studies.

Contact Information:
Authors:
Kartheek Arun Sai Ram Chilla
E-mail: chilla.kartheek87@gmail.com
Kavya Chelluboina
E-mail: ch.kavya2009@gmail.com

University advisor:
Dr. Simon Poulding
Department of Software Engineering

Faculty of Computing
Blekinge Institute of Technology
SE–371 79 Karlskrona, Sweden

Internet : www.bth.se
Phone : +46 455 38 50 00
Fax : +46 455 38 50 57

Abstract

Context. Human oracle cost, the cost associated in estimating the correctness of the output for the given test inputs is manually evaluated by humans and this cost is significant and is a concern in the software test data generation field. This study has been designed in the context to assess metrics that might predict human oracle cost.

Objectives. The major objective of this study is to address the human oracle cost, for this the study identifies the metrics that are good predictors of human oracle cost and can further help to solve the oracle problem. In this process, the identified suitable metrics from the literature are applied on the test input, to see if they can help in predicting the correctness of the output for the given test input.

Methods. Initially a literature review was conducted to find some of the metrics that are relevant to the test data. Besides finding the aforementioned metrics, our literature review also tries to find out some possible code metrics that can be applied on test data. Before conducting the actual experiment two pilot experiments were conducted. To accomplish our research objectives an experiment is conducted in the BTH university with master students as sample population. Further group interviews were conducted to check if the participants perceive any new metrics that might impact the correctness of the output. The data obtained from the experiment and the interviews is analyzed using linear regression model in SPSS suite. Further to analyze the accuracy vs metric data, linear discriminant model using SPSS program suite was used.

Results.Our literature review resulted in 4 metrics that are suitable to our study. As our test input is HTML we took HTML depth, size, compression size, number of tags as our metrics. Also, from the group interviews another 4 metrics are drawn namely number of lines of code and number of , anchor <a> and paragraph <p> tags as each individual metric. The linear regression model which analyses time vs metric data, shows significant results, but with multicollinearity effecting the result, there was no variance among the considered metrics. So, the results of our study are proposed by adjusting the multicollinearity. Besides, the above analysis, linear discriminant model which analyses accuracy vs metric data was conducted to predict the metrics that influences accuracy. The results of our study show that metrics positively correlate with time and accuracy.

Conclusions. From the time vs metric data, when multicollinearity is adjusted by applying step-wise regression reduction technique, the program size, compression size and tag are influencing the time taken by sample population. From accuracy vs metrics data number of tags and number of lines of code are influencing the accuracy of the sample population.

Keywords: Test data generation, comprehensibility of test data, software test data metrics, software code metrics, multiple regression analysis, linear discriminant analysis.

Acknowledgments

The journey of our Master Thesis has been truly an unforgettable experience. We are grateful to work in the software test data generation area. The advantage of both working in the area of test data generation and also performing the Experiment as part of our research method study gives us tremendous sense of achievement. We had the honor to work in the field of software testing while being guided by creative and intense minds in the respective field and the share of knowledge by our supervisor is immense.

We would like to take the honor to convey our sincere gratitude to our supervisor Dr. Simon Poulding. It has been a long journey with lots of challenges, but with his unconditional support and remarkable guidance and trust upon us throughout our thesis made us to reach this point. It is his timely comments and suggestions that helped us to be motivated when we have received poor results in our first experiment and little depressed of it, our supervisor is the one who encouraged us and let us achieve the immense knowledge in the field of our study and this thesis work would not have been possible otherwise.

We would like to thank Kenneth Henningsson for his support for our experiment by allowing the Vinnova students to join our experiment. We hereby take this opportunity to thank our thesis examiner Prof. Jürgen Borstler for his valuable support throughout the course work and his productive guidance helped to complete this thesis work. It is our privilege to thank the Department of Software Engineering for providing us this educational opportunity.

We would like to express our heartfelt gratitude to our parents for standing by our side and supporting us at every phase of our life. We like to thank our friends for their tremendous support and making us cheer when we are really low. We specially thank all the students who participated in our experiment by giving the valid inputs and sharing their knowledge and experiences with us. Their contribution, feedback and reviews uplifted this study. We would like to thank one and all, who we might have missed to mention accidentally, for their unconditional help and support, without which our thesis would have not been successful.

Thank you all,

Kartheek Arun Sai Ram Chilla
Kavya Chelluboina

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Problem Statement

"Investigating Metrics that are Good Predictors of Human Oracle Costs"

Across the industries software testing is involved in every change that is happening around, this makes the testing process even increasingly popular. The software testing is effective as it involves examining the behavior of the system to identify the potential defects [1] [2]. "Overzealous testing can lead to a product that is overpriced and late to market, whereas fixing a fault in a released system is usually an order of magnitude more expensive than fixing the fault in the testing lab [3]". For the software testing to be effective it also depends on the test data that is being used. This means for any realistic software system under test, the test input should be highly structured in nature [4]. The overall effectiveness and the cost that is associated in realistic software system is largely dependent on the type and number of test cases that are used [5] [4]. What happens when an industry designs a new product or review the released systems? To test them adequately they create them into a set of data. This process is called test data generation which is an important part of software testing.

So, the input test data and its generation process are very important for software testing. Test data generation is the process of creating a data set for testing the adequacy of the new software applications. The problem with test data generation is that it is highly complex. There is a major concern in generating realistic test data, moreover realistic test data generation for certain type of inputs is harder to automate so it is more laborious [6] [7]. Thus, despite several advances achieved in the test data generation over the past years the literature show that fully automated software testing is not completely achieved.

Before the test execution the test data should be generated and it requires many pre steps and environment configuration which is time consuming [8] [9]. This test data generation can be done manually or by automated test data generation tools. There is a significant difference between the automation testing and manual testing. In case of manual testing the human interacts with the computer and execute the test cases all manually by himself. In case of automation testing the automation tool is used to execute the test case suits. Once the test is automated the human intervention is not required and they can be run overnight and it increases test coverage. Thus the research interest is continuously increasing in this test automation field and also look into techniques that can cost effectively generate the test data.

The use of meta heuristic techniques to generate automated test data is increasing day by day [10] [11]. The search based software testing utilizes meta heuristic optimization search techniques such as hill climbing, genetic algorithm and many others to automate a task [10]. The main purpose of search based software testing is to generate input, minimize and prioritize the test set [11]. It is a scalable technique used in test data generation, its main objective is to optimize the test data for a property such as coverage, but it doesn't necessarily optimize for other test costs [3]. The very important and most significant area to focus while testing is based on the idea of how to generate the test data that helps not only in identifying high potential faults/strong defect revealing ability but also achieving high coverage [12]. In automated test case generation even though the input is automatically generated, the output must be evaluated with actual outcome intended, hence this makes it a costly process and they reveal and detect only faults and crashes in system but do not tell the correctness of the output [13] [2].

Over the years several advancements are achieved in the test data generation process despite these advancements fully automation is not yet achieved [14]. Any program can be validated by testing, the statements about correctness of the output is stated by various authors as follows:

- The generated test inputs depend on human for correctness estimation [4], for a given input the test oracle is the mechanism that correctly estimates the correctness of the actual output with the expected output.
- For a testing process the main part is the ability to interpret the characteristics of a program, the correctness property can be evaluated [15].
- Software behavior must be validated by human. Generating the inputs for a program is possible but the output must be compared to the input to check the functionality intended is being displayed [16].
- In case if the automation is unavailable it should not be unnecessarily difficult for human to evaluate correctness of the output. The comprehensibility by a human is thus a desirable property of test cases [17] [18].
- A key problem that remains unattended is estimating what rate of the functionally intended is being achieved with the obtained functionality for a given input [3]. Here the point reflects on the correctness of the output for the given input.
- When the test inputs are automatically generated they may be unrealistic to test, one reason to support is unreadability [19].
- Test automation is actually to generate a set of test scripts manually and use a tool to execute over and over this doesn't satisfy the promise of a truly test automation platform [20] [21].

The traditional goal of the automated test data generator is to achieve the structural code coverage [22] [23] only, then what about the correctness the output that is generated? someone must evaluate and compare the expected output with actual output in other words generated outputs should be evaluated to see if it possess the intended functionality. Thus the pass or fail of a test execution which is termed as an oracle problem

is still a major obstacle in the process to attain complete test automation.

Test oracle is the mechanism which estimates whether the software is executed correctly or not for a given test case [5] [21] [9] [24]. The test oracle contains two very essential parts namely oracle information and oracle procedure [25]. The oracle information represents the expected output and oracle procedure compares the oracle information with the actual output [5] [17]. There is support for finding the good test inputs but not focusing on other important problem like cost for checking the output produced for a given test input [26]. Within the testing research there is a belief that there is some mechanism that estimates whether the output obtained from a program is correct or not [27]. The lack of test oracles limits the automation testing techniques usefulness [21]. Given a test input the challenge in identifying the correct behavior from that of incorrect behavior is termed as the oracle problem [2] [28]. For a given input to find whether the corresponding output is correct is a time consuming activity so there is a pressure to make it automated, but generally the automated oracle is non-existent and moreover most of the time it's the human who executes the test cases [4]. So, thus human must check the system behavior and this checking process constitutes to significant cost namely human oracle costs [29] . Human oracle cost is more about checking the output of test cases as to verify whether they are correct [30].

In case of human oracles there is a cost involved as humans are expensive, inaccurate and time consuming so how to handle that situation. So, it is important to identify what are the properties and therefore what metrics affect the human oracle costs. It is important to understand what metrics are affecting the human tester's accuracy and time. Thus we have to understand what factors can help in reducing the human oracle costs.

Research Gap: Search-Based Software Testing SBST describes a range of test data generation techniques that use meta heuristic optimization to find test inputs that are effective at finding faults in software [10] [4]. However, Search-Based Software Testing and other automated test data generation techniques often do not consider whether the test data is realistic and comprehensibility. Comprehensibility may be important if the test engineer need to check that the software's output is the correct one for that input, i.e. is a 'Human Oracle': it will be more time-consuming and error-prone to predict the output for an input that the test engineer finds difficult to understand, in this study we want to better predict human oracle costs

1.2 Research Aims and Objectives

The aim of the research is to identify the metrics that are good predictors of human oracle costs and that can help solve the oracle problem. If we know which metric is a good predictor of human oracle cost, then we can find the trade-off between effectiveness of the test cases and the costs associated in analyzing the test cases.

Given the overall aim, the primary objectives for the research were to:

- To review the literature to identify any work on metrics that can be applied to test data to predict human oracle cost.

- To identify if there is existing literature for the comprehensibility of text/source which in turn are related to the test data comprehensibility.
- Additionally, review the literature for some possible metrics applied on code that could be reused on the test data to check comprehensibility.
- Then next, to use the findings from literature i.e., the potential metrics that could predict comprehensibility and then to empirically test whether they do help in predicting the human oracle costs. To do so, apply regression analysis to identify the correlation and collinearity among the independent and dependent variables, to know if any metrics show variance in time to answer. For accuracy vs metric measure: apply Linear Discriminant analysis to identify the correlations, is the model statistical significance, to know if any metrics show impact in answering questions correctly.

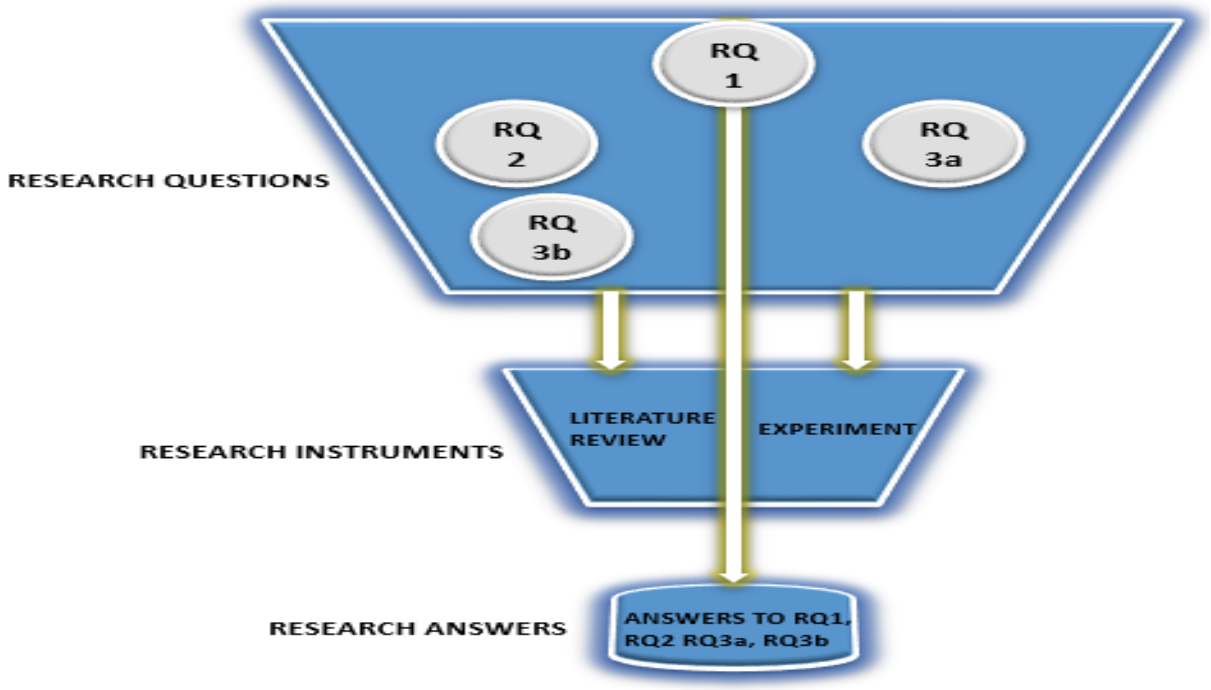


Figure 1.1: Research Instrument

1.3 Research Questions and Motivation

Based on the research aims and objectives, the research question that this study shall answer are addressed in this section. Research question 1 and Research question 2 and RQ.3a are answered by the literature review. On the other hand, the Research question RQ.3b is answered by conducting the controlled experiment within BTH university computer lab.

RQ.1 What are the existing metrics used in the literature, which are relevant to predict the human oracle costs?
Motivation The motivation behind the inclusion of this research question in two fold,

firstly it helps to understand if there is a considerably good amount of literature on the metrics applied on the test data. The metrics selected for the test inputs depend on the type of test data. So, this helps to look more specific to the type of test input rather than all the existing metrics. Secondly, if the literature on the metrics applied on test data. Among the metrics the study look for those metrics that are suitable for the test data.

RQ.2 Are there any existing metrics used in the literature, that can potentially measure the human comprehensibility?
Motivation The motivation behind inclusion of this research question is to understand if there are any metrics that can specifically help to measure the human comprehensibility. If we can identify these metrics that can help to estimate the correctness of the output for a given test input.

RQ.3a To identify if the metrics inspired by source code (code metrics) are usable as good predictors in estimating human oracle costs?
Motivation Code metrics is a set of software measures that provide developers better insight into the code they are developing. So, as we are reducing the human oracle costs in the developer's perspective we would like to look for only code metrics. If there are any code metrics that can be useful to predict then we can take advantage of these metrics and apply them on the test data to check which among these predictors show best significance.

RQ.3b Among the selected metrics that are applied on the test data during the experiment, which of these predictors is/are best?
Motivation The motivation for inclusion of this research question is to understand from the experiment which metric is a good predictor of human oracle costs. This evaluation is done by performing regression analysis to understand which metric is showing the variation on time taken by the subject to answer the test inputs. If the metric shows significant amount of variation, then that particular metric is a good predictor of human oracle costs. The experiment helps to calculate the time and accuracy from the answers submitted by the subjects.

1.4 Expected Research Outcomes

The thesis is expected to reflect the knowledge gained by satisfying the research aims and objectives. This reflection of knowledge is done through answering the research questions. The expected outcomes include:

- Existing metrics that can be applied to the test data and if there is very little literature on test data metrics then are there any existing code metrics that can be applied to test data.
- To gather the metrics that are suitable for measuring the human comprehensibility.
- From the literature review to gather some possible code metrics that are inspired from source code which can be useful in estimating the human oracle costs.
- Any of the metrics selected for the study show statistical significance and show variance in time taken to answer the test inputs and also to know if any of the selected metrics impact in answering the output accurately.

1.5 Structure of Thesis

The thesis report basically consists of four major parts namely introduction, research methodology, analysis and conclusion, as explained in the below figure 1.2. The introduction has three chapters namely Introduction (chapter 1) and background and related work (chapter 2) and Broader view on metrics applied on source code and text. The problem statement, research aims and objectives, research questions are addressed in the introduction. The background and related work (Literature Review Methodology and Results) reflects more about the applied research method for literature, test data metrics. The chapter 3 is more about code metrics, comprehensibility of text and metrics selected for experiment. The research methodology has primarily two chapter namely the research method (chapter 3) and the experiment setup, execution (chapter 4). The research method is about the type of research method applied in this study. The experiment setup and execution is completely addressed in chapter 4.

The analysis (chapter 5) performed on the experiment results is addressed here. The analysis section is to perform analysis on gathered data from experiment results. Finally, the conclusion section which is divided into two parts namely discussion and limitations (chapter 6), discussion is about overall results and the limitations. The conclusion and future work (chapter 7) presents the summary of the contribution from the research study and the future scope for expansion.

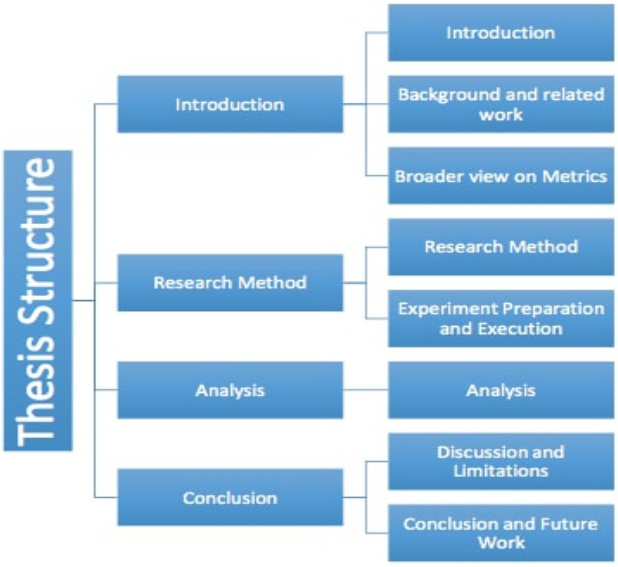


Figure 1.2: Thesis Structure

Chapter 2

Literature Review Methodology and Results

To better understand the current research, the first important and essential step is to understand and analyze the existing metrics that are applicable for the test data comprehension. So, this chapter 2 address the Literature review methodology applied in this study. In addition, what are the existing metrics that are available in literature which can be applied for comprehensibility of test data.

2.1 Literature Review

As per the guidelines given by Hart in [31], literature review is defined as "the use of ideas in the literature to justify the particular approach to the topic, the selection of methods, and demonstration that this research contributes something new". It helps to create a firm foundation for advancing knowledge in the area of research. It helps researchers to clearly understand the existing body of knowledge. Authors of [32] proposed a systematic approach to perform literature review. The authors in [32] define literature review process as "sequential steps to collect, know, comprehend, apply, analyze, synthesize, and evaluate quality literature in order to provide a firm foundation to a topic and research method". Finally, the output of the literature review process should be able to demonstrate that the research that is proposed contributes something new and useful to the overall body of knowledge [32]. The process through which literature review can be performed is clearly shown in the figure 2.1 below.

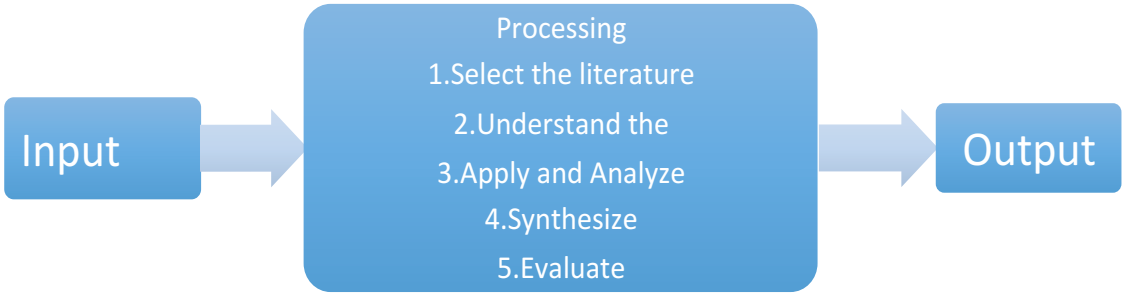


Figure 2.1: The Process illustrating how Literature review is being conducted.

In order to execute the process of literature review, we have selected snowballing as our sampling approach in order to perform the literature search. It is mainly aimed to filter the literature in order to improve the quality of our research and further, the selected literature is carefully analyzed and useful data is extracted from it. Snowballing

procedure that we followed for this thesis is clearly explained below.

Why Snowballing is chosen as a search approach?

In this review we had chosen snowballing as our approach for literature search. This method is stated in [33], it briefly explains about the guidelines to perform literature reviews. It clearly specifies the techniques of using the citations and references that relates to a particular paper and after that identifying the further relevant papers. Quite in many cases it is very straightforward to find and identify the relevant papers and reduces the probability of missing out the related papers [34].

We have many researches that emphasize on lack of the research on this specific research field. Thus the author has been opted for further agile approach to find out the related literature instead of the traditional approach, where database approach is used. While performing the database approach there is a probability of missing out some likely relevant articles. There may be various factors for this. One of the crucial and most occurred factor is the trouble in formulating the appropriate search string with the terminology. There is also a possibility of getting numerous number of irrelevant papers, in case, if the search string consists a general view point [33]. In paper [33] few examples had illustrated, showing that some papers are retrieved with the help of snowball approach instead of the database approach. The reason behind this came out after this argument has been examined and found that the inconsistency while choosing the terminology, affects the search string. Each and every approach will have its own advantages and the selection is done by keenly examining the intricacy of the research that is being conducted [33]. Hence here in this situation, the shortage in literature and then for some papers that are previously retrieved, the view and concepts are not direct and forthright which made to do a deep examination. Taking all these issues into account snowball approach is chosen which validates to be constructive and helpful for this study than the database approach.

Database used for finding the Tentative Start set of papers?

The database that is recommended to find the initial set of papers for snowballing is Google Scholar [33]. The specific benefits for taking this database is stated in [33] it helps to overcome the complexity in publisher bias and problems to access the papers. It also has few drawbacks. The huge amounts of records are retrieved which makes hard to find the appropriate set of papers. Kinsley explains the benefits of using "Inspec". [35]. It is also explained that why "Inspec" is considered first than "Google Scholar" to search the papers. Both Kinsley Charles and Kinsley Karin conducted a study about to find the required information in an effective way. While undergoing in search process it is a part that databases are compared with the consolidation of the results that are being retrieved. Engineering Village makes it to choose first than Google scholar by providing additional features. By taking both the benefits and drawbacks of the "Inspec" and "Google Scholar" into account, the author had used later to form the initial set of papers, since this study was recommended consistency in terminology, and to which range the articles must be searched. Thus Engineering Village is chosen as the primary database, and Google Scholar is taken for the secondary database.

2.1.1 Snowballing Procedure

Wohlin in [33], described the procedure of snowballing in four steps that include Start set, iterations, authors and data extraction.

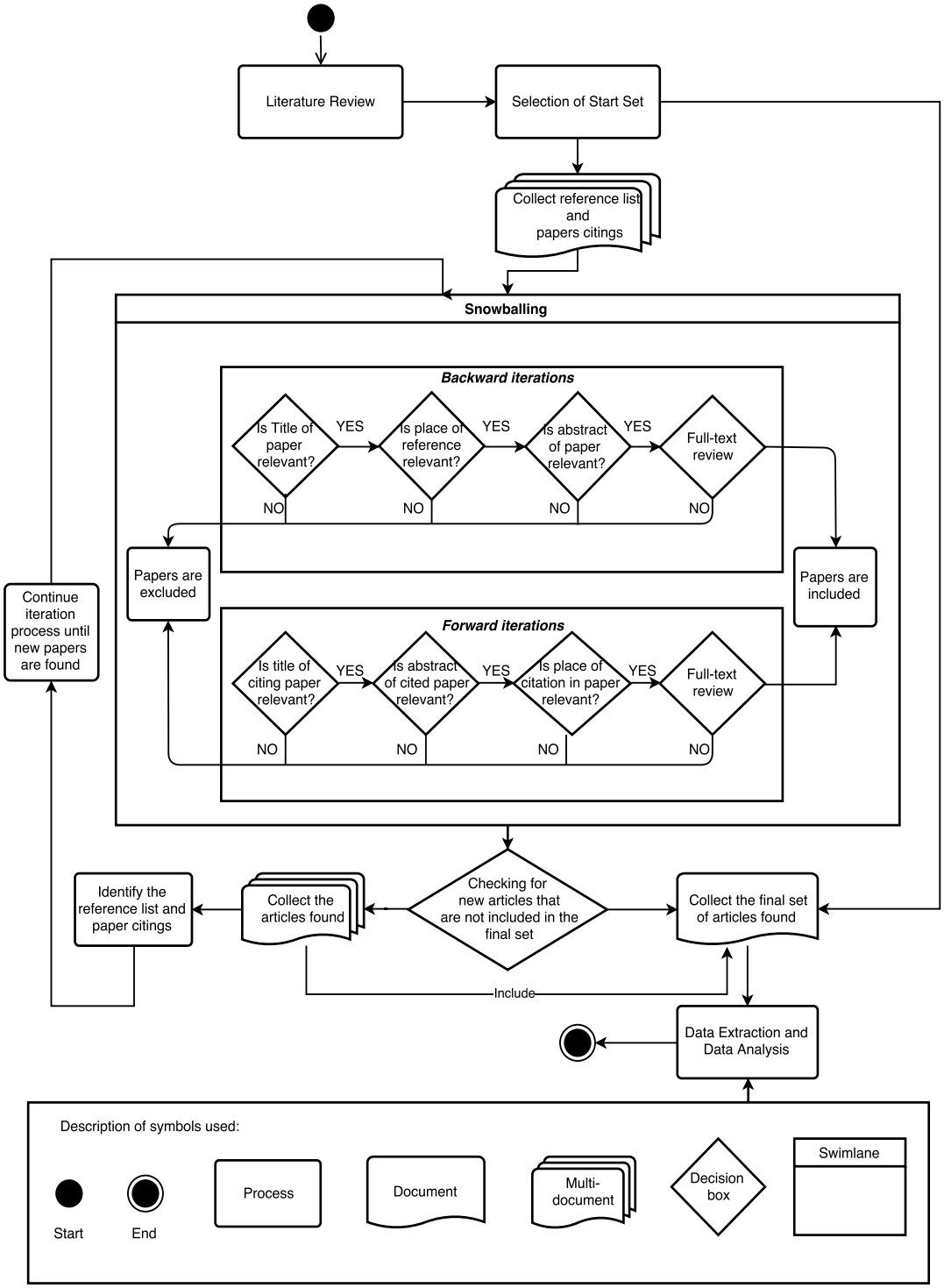


Figure 2.2: An overview of research methodology

Initially, appropriate search strings should be framed that give better results related to the selected area of research. After performing searches in the selected databases, start

set articles should be identified. These articles should reflect the useful information of the current research gap. After finalizing the start set, backward and forward iterations are performed on the start set articles. Backward iterations are to be performed by observing the references of the start set articles and the forward iterations are to be performed by observing the citations of the start set articles. After finalizing the articles obtained through all iterations, data should be extracted by carefully going through each article. The entire procedure of snowballing is shown clearly in the figure 2.2.

Start set keywords:
Firstly, we need to get the Start set of papers, to achieve them we have to identify some right keywords. The key words are usually identified from the research questions. The keywords that we have taken here are listed below.

Human Oracle costs Oracle costs	Test data generation Automated Testing HTML test inputs Html test data
Software metrics Code metrics	Comprehensibility of test data

Table 2.1: Keywords

Search String:
As soon as we identified the keywords, we have to formulate search strings. These search strings are used in the selected databases to gather the articles. We used combinations of various Boolean operators in the search string, to collect the most significant and relevant articles.

The search strings we used here are:
Set 1: (Human oracle costs) OR (automated testing) AND (software metrics) 419
Set 2: (Human oracle costs OR oracle costs) AND (software metrics OR test data generation OR automated testing) 80
Set 3: ((Html test inputs OR Html test data AND Software metrics AND code metrics comprehensibility of test data OR human oracle costs) 34

The database that is chosen to carry out the snowballing is INSPEC database. To achieve an appropriate start set of papers related to the study and formulating the search string are both very crucial and challenging steps in the snowball approach.

Start Set
All the related articles suitable for the study are gathered and the necessary steps for the snowball sampling are performed. Here we have selected inclusion and exclusion criteria for the study depending on the research questions and to get the most relevant papers. By the help of the search string we obtain numerous articles in which most of them are not related to our research area. Hence an inclusion and exclusion criterion is applied and excluded all the irrelevant articles from all those numerous articles. The inclusion and exclusion criteria are briefly described below. The start set with all the articles are presented in Appendix figure E.6

Inclusion criteria:

- Articles available in English.
- Articles published between 2001-2016
- Articles which are peer reviewed
- Articles with full text availability
- Articles that mainly focuses on the metrics
- Articles with related abstract of the study

Exclusion criteria:

- Articles that focus on further topics rather than research area.
- Articles that are repeated.
- Articles that does not show proper outcomes.
- Articles that does not satisfy inclusion criteria are excluded

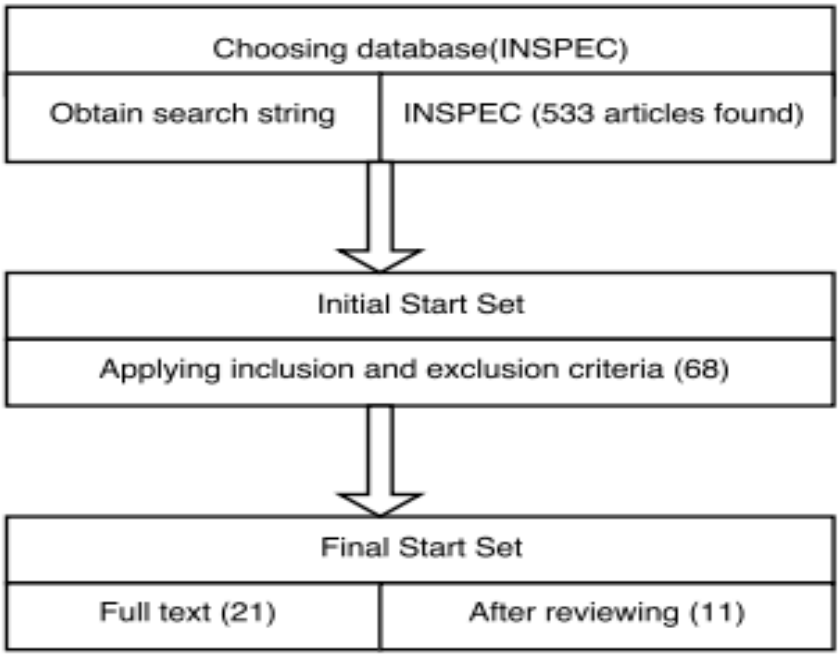


Figure 2.3: Start Set

2.2 Software Metrics applied on Test Data

Metrics are useful in measuring a product or service [36] [37]. "Software metrics and measurement are both interrelated, software metrics describes wide range of activities concerned with measurement starting from producing numbers that characteristic properties of source code these metrics are called as classic software metrics to models that describe software resource requirements and software quality [38]." Metrics are always overhead on software projects typically it would be around 4-8 % [39]. Software metrics vary for different technologies that are used, type of programming languages [40].

Impact of Design metrics over fault proneness: Software metrics possess a great value of information that can help in software quality prediction during the software development process [40] [13]. The impact of the CK metrics metric suite to identify the fault prone systems is analyzed by Basilli.et.al [41]. In the below table 2.2 some of the classical metrics applicable.

Halstead Program	Length, Volume, Level, Difficulty, Effort and time required for Programming
McCabe	Cyclomatic, Complexity
Miscellaneous	Branch Count

Table 2.2: Summary of CK metrics suite applicable to object oriented design explained by Chinadamber and Kemerer

To measure the re usability of patterns four metrics have been described 2 are related to comprehensibility [42]. The metrics help organizations to generate effective websites this indeed provides measures for managers to understand and replicate [43]. For the success of the website factors like frequency of use, information quality, user satisfaction are all elements [44] [43].

1960's is the decade, when the software metrics first came into picture during this period Lines of code is applied as a measure to predict both programmer productivity and program quality [39]. The lines of code is one of the measure of various notations of size such as complexity functionality and effort [45]. In the early 1970's the drawbacks of Lines of code as a measure of different notations size are identified [46]. Different languages have different notations, schematics, Formal automata state notations. For example,

- The depth of the tags within the HTML now is different from the depth of inheritance of scripts when it comes to Java script.
- An lines of code in an assembly level language in terms of functionality, effort, complexity is not comparable with an LOC in high level language.

Defects for Lines of Code is used for measuring software quality, it acts as a means for assessing productivity [39] [45] [47]. Luchscheider et al. [48] says to evaluate and prioritize the test case models the common metric average percent of faults detected can be useful. The defects in operation level which are termed as failures are different from the

defects that occur at development level which are termed as faults. Faults may or may not lead to failures [39]. No of defects is a good predictors of the quality of the website [49].

In case of FORTRAN languages, the measures to estimate the programs quality are Program length, program level, program difficulty, program volume, program effort and program bugs, Cyclomatic complexity, Source lines, source lines comments [44] [50]. The McCabe's Cyclomatic complexity is extremely popular among complexity measures and easy to calculate using statistical analysis [30] [39]. The metrics like depth of tree and number of child nodes for each class are useful in measuring the HTML [51] [52] [53]. The purpose of above metrics is to identify if they are good predictors of the fault proneness in classes.

A relationship between quality factor and the development metric are illustrated as follows: Reliability: known errors, understand ability: how complex is the code, Modifiable: time to fix known errors and Correctness: Modification requests [54]. The reuse can be applied to functions and modules that are within the programming languages [55]. The firms with CMMI level-5 set benchmarks throughout the organization for key project metrics like productivity, profitability, in process quality and conformance quality [56] [57].

Lines of code is a simple measure for a program [52]. Halstead metrics are based on his work in Halstead software science, his work primarily measure program size, complexity, program level and volume [46] [58]. McCabe's prefer an abstract representation of Control flow graphs and also best known for Cyclomatic complexity [46] [48].

It is hard to evaluate every line of code that is programmed [58]. A case study supports static analysis is useful to uncover the properties in a program, the static analysis is helpful for both students and examiner to understand the program [58]. Mengel et al. [58] explains metric like number of operators, number of operands, number of statements, Cyclomatic complexity can be useful to measure the size and complexity of a program [58].

Khoshgoftaar et al. [59] supports unlike previous view on software metrics where more the number of lines of code, more complex program which in turn has more errors. Over the year's metrics evaluation is beyond simple measures and Luchscheider et al; O. Signore and Jiang et al. [48] [49] [60] supports the importance to find the correlation between the metric when applied on different complex models.

2.3 Related Work

To conclude from the above literature we found size as a measure that can be applied to test data. Size is in character bytes and can be applied to any type of programming language. There are other metrics applied on test data from above literature in 2.2 however,they are not promising because the metrics proposed are mostly referring to object oriented paradigms. As we don't want to understand all the software metrics which could be used, it is to understand of the specific field rather than complete understanding. So, given this situation i.e., as no promising metrics are found we have changed our initial plan on finding metrics relevant to test data into a broader perspective to start looking for metrics applied on source code and text.

Summary of the Findings: We did the literature and we found that size metric can be applied for test data comprehension. In a broader way this one metric is not sufficient to conduct the experiment so we have to further enhance the literature study. Since from this literature we didn't find so lets look in chapter 3 a much broader view of other metrics which might be relevant and this is where we have looked into metrics that can be applied on source code and also on the text. If we find any relevant metrics then we can take some of these possible metrics applied on source code and text and use in this study.

Chapter 3

Broader view on Metrics applied for Source Code and Text

Initial plan was to review the literature on test data we did not find anything really good apart from size lets now have a look into other broader views. This chapter 3 gives a broader view on metrics applied on source code and text. Since we only found size as a relevant metric from chapter 2. Only size is not sufficient for conducting the experiment so we have extended our literature in a broader way into source code and text comprehension metrics.

3.1 Code metrics that can be relevant to the experiment

There are several categories/types of metrics like design metrics, code metrics,quality metrics and so on however we chose only code metrics because they are from developers perspective. Code metrics is a set of software measures that provide developers better insight into the code they are developing. So, as we are reducing the human oracle costs in the developers perspective we would like to look for only code metrics.

What metric to choose depends on programming language: Walker et al. [61] supports the argument that there are many languages and techniques under the roof of programming. For our study we would like to consider HTML as our test input. The HTML in terms of protocols and the implementation advancement is increasing for the past three years [62]. HTML has become important part and parcel of the web development itself [63] [64]. As we did not find any metrics that are relevant to test data so started to look into literature for software code metrics which can help to relate to comprehensibility understand-ability of test data.

Some metrics are useful to measure the plagiarism in websites this normally arises due to the copy of content [56] [65]. General strategy while designing web applications is the developers design the initial pages and reuse the code in initial pages and apply them to next once [65]. So, each page is considered as control component of each actual page created from this template and the added information is nothing but the data component of the that page [65]. The main reason to explain about website, the Di Lucca [65] used HTML tag as a measurement to analyze the code clones in client side static web pages [66].

The website is more than a single page application thus it contains the page links.

These page links are of several categories like the inner links (number of links going in the page itself), the number of outer links (links to next page within the website) and external links (links going to other sites) [49].

Kitchenham et al. [67] discusses about some of the code metrics like Size in Lines of code [37] [68] and branch count. The author supports that code metrics were better to identify complex programs, change prone and error prone than design metrics this is performed to understand usefulness as results say that correlation exists between code metrics and known errors complexity of the code [67].

Software science try to quantify the metrics like size and complexity that are normally addressed as the fundamental set of measures [50]. Poulding et al. [29] believes the comprehensibility of a test case is very important for a human. In this case finding faults and bugs in the program is not their objective but to understand trade-off between coverage and comprehensibility [29]. Use of programs as test inputs is more feasible than using the grammar because programs can enable structure constructions and also can store value which is not possible in grammar [29]. To achieve high code coverage single input test case with large XML input is more suitable [29]. Quantities that effect the comprehensibility of a human in measuring correctness of XML test input are number of elements, number of attributes and number of nodes [29].

Lucansk'y et al. [69], states web page contain easily process able mark ups, these mark ups can be evaluated using the automatic term recognition algorithm this algorithm is applied on the HTML tags present. The alphabetically ordered list that is visible when a letter or word is typed in a web browser can be modified by changing the features within title tags, meta tags and apply keywords in URL, thus tags are very important in the HTML [70].

3.2 Broader View on Comprehensibility of Text/Source-Code

This section take a broader perspective on comprehensibility of test data to know about the work done in analyzing readability and understanding the source code/text. We have briefly look into measures for code comprehensibility and text readability/comprehensibility as both are strongly related to test data comprehensibility. Biggerstaff et.al [71] given a formal definition for program comprehension which is as follows "A person understands a program when able to explain the program, its structure, its behavior, its effects on its operational context, and its relationships to its application domain in terms that are qualitatively different from the tokens used to construct the source code of the program [71]".

For source code readability both structural aspects (line length, number of comments, looping statements, number of spaces) and textual aspects (code within identifiers and comments) play significant role in program comprehension and software quality. Both structural and textual features together improve accuracy of code readability [72]. The elements like source code design, formatting and visual aspects impact the program un-

derstanding [72]. For better readability and comprehension of source code enhancing the syntax and semantics of the program using methods like standard generalized markup language can be done [73].

To increase the readability and improve the understanding of the program code some program guidelines often include formatting standards like indenting loops and conditional branch statements [74]. Xiaoran Wang et.al [74] and Andrea De Lucia et.al [75] supports that code's size complexity and readability are influenced by the identifiers names, appearance and comments. The complexity and comprehension are affected by the duplications in source code [76]. Majority of the source code text is influenced by the programmer defined identifiers and these identifiers heavily depend on the readability and comprehensibility of the source text [77]. It is better to create a common starting set of identifiers names before designing a new system to avoid overlapping. Abbreviations should be different for different words and are to be consistent throughout the source text [77].

A source code is comprehensible when a new developer can understand and implement changes to the source code quickly and reliably. For the team to effectively be scalable the code needs to be comprehensible before it is modular, reusable, testable and reliable. Some important ways to improve the code comprehensibility can be by using following steps [78].

- Write the source code from reader's perspective, which means even the developer can perform modifications quickly.
- Try to avoid duplicate code patterns and long methods this is susceptible to bugs.
- Define clear ownership and responsibility of each function module and components can help reduce code incomprehensibility.

Hanspeter Mossenbock et.al. [79] argues that active text in particular the hypertext can be essentially very useful in understanding and structuring the code, as the programs are read selectively unlike sequentially. For structuring the code several features have been useful for several years, these features are namely Folding which helps to replace/-collapse the code with shorter text this can be applied in loop statements, for example if the original code can be replaced with shorter code then the number of lines and depth of the code vary which indicates the depth of the source text changes.

Kazuki Nishizono et.al [78] used a small Java application and performed modifications in the source code, the consistency of code comprehension strategy and comprehension effort estimating metrics like lines of code are used to assess the time taken by the participants to assess the modifications done on source code. The results show that comprehension metrics and strategies are not consistent with different modification tasks.

Jonathan Elsas et.al used the TTR Table tag ratio which is the estimation of total number of table tags to the tags in the HTML document to classify the web pages. The final results support that the use of HTML tags in the Hypertext documents is quite rich and modular, he supports that much more information can be learned by analyzing the use of HTML tags.

Along with variables like commenting, blank lines insertion and control flow the program indentation is also an important factor for program comprehension. After applying both blocked, unblocked and four levels of indentation ranging from (0,2,4,6) spaces the author concluded that only some indentation (2,4) spaces show highest mean value for program comprehension [80].

Text comprehensibility can be improved by invoking multiple self-selected feature options like color, photographs, video, graphs, hypertext and hypermedia. Marshall [81] argues that readability and text comprehensibility cannot be sorted out using readability formulas as the formulas do not measure meanings. Text comprehensibility is a primary concern and research studies for ensuring optimal match between reader and text is a concern in world of computer technology. Adéline Astrid Bourbonnière [81] used the integrative inquiry approach to find factors that influence the comprehensibility of hypertext and hypermedia. The so called "outside the head" factors like separate, movable, overlapping windows, intensive electronic environment, navigational aids and comprehension monitoring options. inside the head factors include prior knowledge of navigation procedure.

.

Filippo Ricca et.al [82] tested the websites comprehensibility using keyboard based clustering by converting the websites into graphs then the participants are requested to examine the websites. The author supports as the size of the website increase the complexity involved and the graphs complexity and design also increases. Rudi Cilibrasi et.al [83] describes the source code is also in the form of text and sometimes there is a lot of repetition of text, feature based similarities white spaces and similar kind of code repeating multiple number of times, this influences the overall size of the document. It discusses using compression to calculate a similarity distance metric, motivated by the fact that the compression size is an approximation of Kolmogorov complexity, and therefore the “information content" of a piece of data.

3.3 Related Work

we performed an extensive search beyond the test data metrics as the literature is considerably low so we looked for some possible code metrics that can be applied on test data. There are many metrics out there it is very important, in this research to identify and look for those possible metrics that might be related to the comprehensibility of test data. There are some common metrics or generic properties like size compress size measures that is used for all programs both in chapter 2 and chapter 3 Size is applied irrespective of programming type. Many acronyms are available to measure the size for example lines of code [37]. We found some metrics like number of tags and depth of the tree nodes these two metrics are noticed in the literature and for this study we considered they might have some potential impact so these two metrics are taken into account.

The source code is also in the form of text and sometimes there is a lot of repetition of text some data resembles same/alike, feature based similarities white spaces, this influences the overall size of the document. The compression size is an approximation of Kolmogorov complexity, and therefore the “information content" of a piece of data. So, the compress size is different from the size and it is always lesser in bytes. The compress

size can be applied to any programming language irrespective of type.

The comprehensibility of the source code is influenced by the depth/ level of the source code. Writing a code which is readable, reliable and understandable to existing developers and new developers who would like to reuse the code is very important. This process of writing code involves defining identifiers and several loop statements, however writing them in an efficient way with lesser duplicates could influence the depth of the source code, thus to measure the comprehensibility of text in source code depth can be one metric that can be applied. From the literature we found that three important metrics influence the comprehensibility of test data/ source code they are Tags, depth of the elements in source code and the compression size of text.

By considering the broader view of comprehensibility of test data we observed that the metrics like depth of the source code , compression size of the text and the tags in HTML does have influence on comprehensibility of source code and text. This argument is supported by the literature addressed in the section 3.2.

3.4 Summary of the findings

We observed the literature to select some possible metrics relevant, there are many metrics out there but we have only selected some possible metrics for this study. Interestingly, both the literatures performed on text data comprehensibility and the code metrics strongly support that the depth of the source code does influence the readers ability to comprehend the code. Tags in the Hypertext documents is quite rich and modular, more information can be learned by analyzing the use of tags. The source code is also in the form of text and sometimes there is a lot of repetition of text, feature based similarities white spaces etc, this influences the overall size of the document. The compression size is an approximation of Kolmogorov complexity. So, the compress size is different from the size and it is always lesser in bytes. The compress size can be applied to any programming language irrespective of type. So, from the literature we found four metrics, there are several other metrics but we selected only some possible metrics that can be applied on test data input.

How the metrics are used: The metrics selected are calculate for each test input (Test inputs are addressed in chapter 4) and if and only if the test input examples show significant variation only the those test inputs are taken into account for experiment. The duplicate test input examples are avoided, the metrics variation is very important as the statistical test will be used to identify if any of the metrics influence the ability to identify the correct output for a given test input. This in turn can help to predict the metrics that influence the human oracle costs. The calculation of each metric for a given source code is explained in the chapter5.

Chapter 4

Research Methodology

Software engineering makes use of mainly two types of research methods such as qualitative research and quantitative research [33].
Qualitative research: : This can be referred as exploratory research where the focus is to study the objects and observe the findings in its natural environment. For example, literature review is a part of qualitative research study.
Quantitative research: This can be referred as explanatory research where the focus is to compare tow methods, processes or techniques in order to identify the cause-effect relation between them. Such type of study is conducted through a setup rather than a natural one. For example, a controlled experiment is a part of Quantitative research study.

An overview of empirical research methods that are commonly in practice [84]:
Survey: "A survey is a process of collection of information from or about people to understand, compare or explain their behavior, attitudes and knowledge." It is a retrospective investigation where the both qualitative and quantitative data can be retrieved through questionnaires and interviews. In such study, a sample population is considered to generalize their results later to a larger population.
Case study: "It is an empirical research method that relies on multiple sources to investigate an instance or number of small instances within its real context, especially when the boundary between context and phenomenon is not clearly specified." It is an observational study where data collection is done throughout the process.
Experiment: "An experiment is a controlled study conducted by manipulating a factor or variable of the studied setting." Measuring the effect of variables while making some variables constant by applying different treatments to different subjects based on randomization is called an experimental procedure.

The survey is not suitable for this study as the study is not collecting information to describe, compare and predicts attitudes, opinions, knowledge and behavior [84]. Since the study is not either observational or exploratory a case study is not suitable. So, as the study is investigating the casual relationships among the study variables the experiment is a more suitable for this study.

Before performing the experiment a literature review was performed. Because, the literature with respect to current study is very low and applying other methods like systematic literature review or systematic mapping study would not be suitable as our study does not have much literature to start performing these methods. Since Literature review using snowballing is feasible to know both in forward and backward searches on the start set to understand the totality of set.

In our research, we conduct a literature review followed by a controlled experiment in order to answer the formulated research questions. Our research deals and primarily focus around the experiment so we tried to avoid applying all the other research methods because they do not suite this study. However, another way to perform this study could be possible through an industrial case study but as such possibility is not possible due to unavailability of such opportunity to work within an industry. So, we performed the experiment using the University Master thesis students from the department of Software engineering as our sample participants. Sampling of the participants is done based on whether they have experience in HTML, if they do have experience then they are requested to kindly participate in the experiment.

4.1 Experimental Design

"An experiment is an empirical research method that investigates the casual relationships and processes [85]". It is conducted to obtain a direct and a systematic control over a situation by manipulating its behavior.

In general, there are two types of Experiments:
Human-oriented: It involves humans who apply different treatments to different objects.
Technology oriented: It involves the application of different tools to different objects.
In our study human-oriented approach is adopted.

To conduct a controlled experiment effectively, the activities and concepts to be defined are [85]:

Experimental Design: Wohlin et al [84] explained the process to design and conduct an experiment in software engineering. The recommended experimental design is based on statistical assumptions made along with the selection of subjects, objects, instrumentation and other factors to conduct an experiment.

Variables: The objective of a formal experiment is to study output when there is a variation in the input variables. In general, there are two types of variables

- Independent variables: When a variable can be controlled and manipulated, then such a variable is called independent variable. There is a total of 8 independent variables observed in our study such as size, compress size, number of tags, depth of the node, number of lines of code, tag, anchor<a> tag and paragraph <p> tag.

- Dependent variables: A variable, which is not affected by a change done in the process, is called dependent variable. Usually there is an only single dependent variable.
  In our study, Time is the dependent variable.

Treatment: "A treatment is one particular value of a factor." Factors are the variables that undergoes a change i.e., independent variables in an experiment.

Subjects: "The people that apply the treatment are called subjects". In our study master's students with intermediate to expert level knowledge in HTML coding are selected

as subjects.

Object: Object is the medium or programs that is needed to be reviewed or inspected. HTML test data input is the object in our study.

Instrumentation: The tool used for conducting the experiment in our study is SPSS statistical program tool. The SPSS statistics program tool is used in this study to perform the regression analysis. Several important things are considered before conducting the final experiment. To reach the final experiment there is a step by step procedure that we implemented this entire procedure is an experimental protocol which we believed would help to reach the final experiment. The experimental protocol main aim is it has small number of goals that are to be reached firstly before final experiment is conducted. Then after the experiment is conducted how to analyze the results the entire scenario.

There are several other tools that can be applicable other than SPSS statistical tool, But this is very simple and easy to comprehend, data analysis is easy, easy to post the data and analyze. Other tools like R they involve some programming to perform analysis so tried to avoid those tools. So, we selected SPSS statistical tool as it is easy to perform analysis over Excel spreadsheets and R tool. We only applied regression analysis because here the dependent variable is time which is continuous variable and independent variables are continuous as well so regression is a suitable technique.

For Literature review we only considered literature using snowballing because the number of articles specific to our study are relatively low so we neglected systematic mapping and only used snowballing for literature.

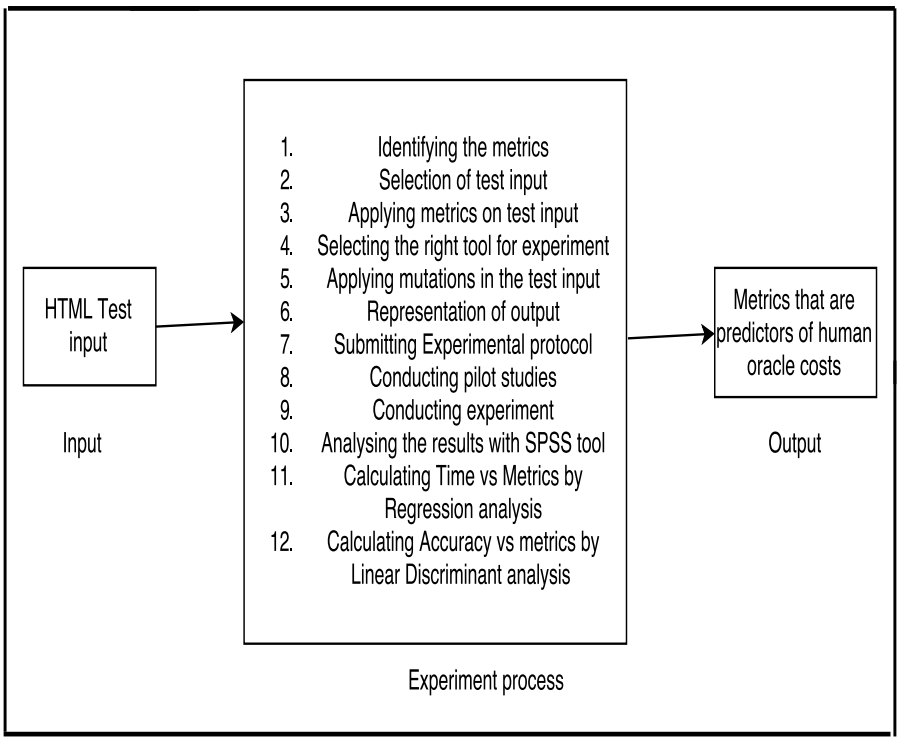


Figure 4.1: Experiment Design

The group interviews are considered over other interviews because already the participants spend more than 90 minutes of time during the experiment so again requesting them to participate in the interview individually will consume a lot of time. We want to conduct just after the experiment is finished so we have no other better solution than group interview.

4.1.1 Experiment Procedure

1. How do we conduct the experiment?
   Before The experiment:
   - Send the mail to the participants asking them to register for the experiment on the particular date and when they are free to appear for the experiment.
   - Lime survey hosting helps to send the invitations for the experiment and also send the reminders as well.
     * Email Invitation for experiment
   - Experiment is conducted to knowledgeable person only, that means a pre questionnaire is needed to be filled by the participant who have experience in HTML.
     * Fill the pre questionnaires link sent in Email invitation.
     * Reminder will be sent to the participant on the day of the experiment.

   During the Experiment when the participants arrive to the Experiment Lab:
   - When participants enter the instructions are given about the experiment.
   - The instructions page is given to the participant about the input and also the time logging instructions.
   - The participants are given the input and they check whether the single output is correctly matching the input.
   - Recording the time is automatically done by the Lime survey.
   - Since The experiment is conducted in a controlled lab environment,Pike up a room that is required for conducting the experiment.
   - A fixed time of 1 hour is set in the software lime survey so that all the participants start at the same time and finish at the same time.
   - Even if they are unable to finish the experiment in time the section stops and saves the data which is answered by the participant irrespective of completion
   - We are going to randomize the test inputs which is done using lime survey.
   - The participant answers the question in a serial order they cannot skip to the next question until they select one of the three multiple choice questions.

   After the Experiment:
   - The participants are given the post questionnaire to address the challenges and recommendations about the experiment.

Between the post questionnaire, during and after the experiment there is no break and this entire process is continuous. This enables to give the feedback by the participant right away and avoids the impact on feedback.

4.2 Area of Study

Software testing has been and continues to be vital software engineering area of research. The research being conducted over several years in the areas like the test data generation, automated test data generation and human oracle costs. There has also been empirical research study in the areas of test data generation within the context of human oracle costs and the measures to avoid these costs. The automated test case generation and therefore search based software testing are good indeed because they reduce the costs or increase the quality but that is not the only costs. This study takes primarily about other costs that is the costs for running the test cases in particular analysis of the results which has to be done manually. This manual analysis is performed by human so the correctness of the output for the given input is analyzed by human so conventionally to understand what makes the test data hard or easy to understand by the human is the primary area of focus in this study. The area of study is explained in the figure 4.2.

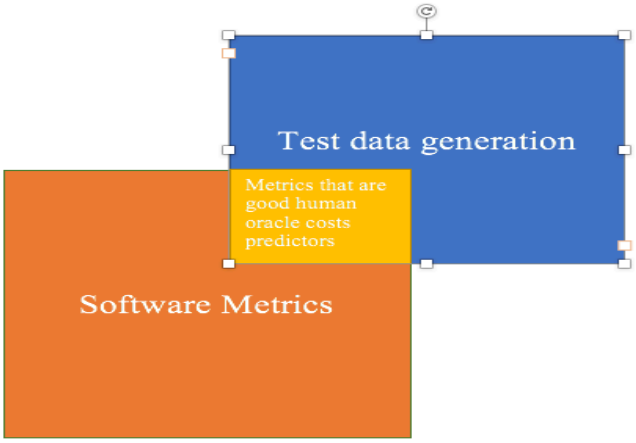


Figure 4.2: Area of Study

Chapter 5

Experiment Preparation and Execution

In this chapter 5 we focus more about preparation and execution of the experiment. Controlled experiments in the Software development field require a great insight for planning and care to attain a meaningful and useful results [86] [87]. The usability of the results always depends on the careful experimental design [86] [88].

5.1 Final conclusions on metrics selected for the Experiment

Selection of Metrics: This section contain the metrics obtained from the literature. The Metrics that we chose are Size, Compress Size depth and Number of tags for test input.

5.1.0.1 Size

We use size as a measure for the test input [37]. The size of the program is represented in number of character bytes. We have chosen 19 test inputs, the representation is done in bytes because when comparing the size of a test input file with the compressed size in kilobytes for the same test input file it doesn't show much of a difference. Whereas when represented in the byte's format then there is a considerable amount of difference. Conditions applied for selection of test input are size should not be very small and test input can be compressed significantly.

5.1.0.2 Compress size

Compress size is very simple to perform it is metric to understand to what extent the file is compressed, it help to looks at diversity in test data [83] [89]. One cannot compress random characters like ABCDEFGH and so on but if it is not random character like AAAAAAA or BBBBBBB then we can compress them. During compression the repeated statements, white spaces are compressed to decrease the size of the test input. We would make sure that there is a perfect collation always so that there is a significant amount of compression done on the file. To do so we need to perform best compression technique applied on the files this can be possible when the below command is typed on the file,

In Command prompt type: Gzip Filename –best

Before Compression:

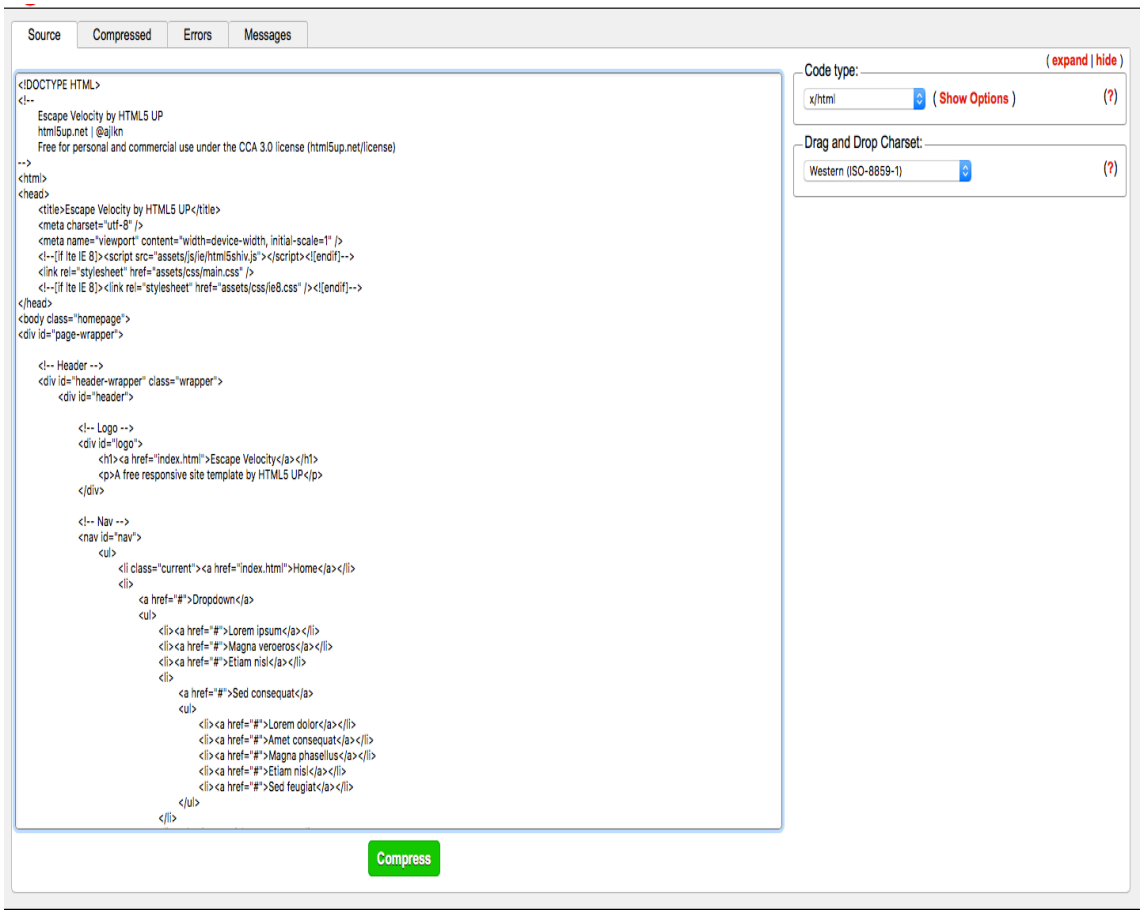


Figure 5.1: Compression tool that helps to compress the HTML test inputs original test input without compression.

The internal compression algorithm does the work to attain the maximum compression. It is so profound in terms of compression that looks for all the possible opportunities that help to compress the file. For example, the test input has 200 lines of code with same tag repeating itself then the compression algorithm shrinks it to a smaller size, its work over the tags in the test input is highly efficient. To understand much clearly we can compare two outputs one source file and the other Compressed once with no white spaces.

After Compression: The file compression is highly useful to understand the internal structure of the program space occupancy. The rate of compression in above case is 72 % as we can see in figure 5.2 it saves 2978 byes from the original 10806 bytes. This is because it looks for things like white spaces, comments, breaks and try to avoid them in the compression. Question is if it is understandable to the human or its hard in terms of coverage with respect to normal test input.

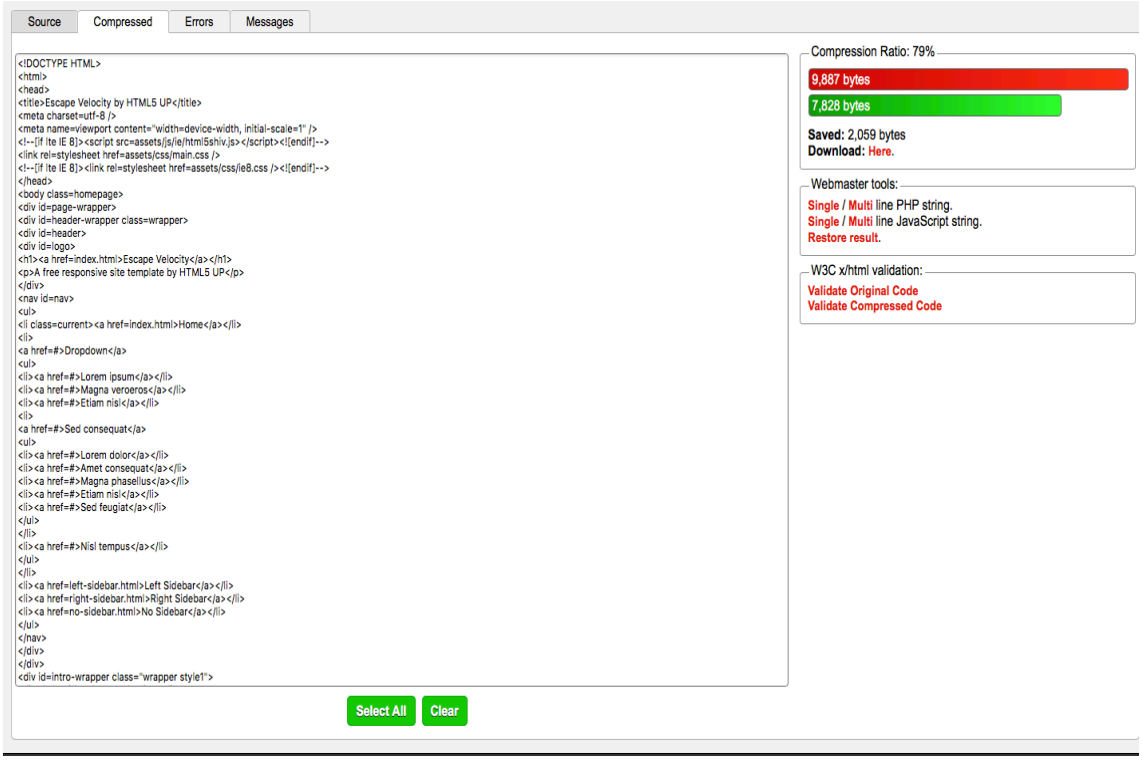


Figure 5.2: Compression tool that helps to compress the HTML test inputs, original test input after the compression is performed.

5.1.0.3 Depth

HTML is a tree like structure so there is a depth involved in terms of nodes [90] [91]. The depth is a specific to file, the depth below figure 5.3 is 7. This is one way of representation when the parent node is starting from 1. In other words, depth is nothing but the maximum number of parent traversals that are needed to reach the root of the tree [91]. The depth is measured as a whole for the entire document or file. This traversal should account for reaching root from any node in the document. So, it always looks for maximum depth or how deep is the test input and present the data in terms of numeric.

Figure 5.3: Illustrating the depth of the node, as the count increases the depth of the node increase.

5.1.0.4 Number of Tags

The number of tags count is equal to total number of tags that are present in the source code [92]. For example there are different types of tags that are available within the HTML like Heading tags<h1>; Line Break
; Phrase tags; meta tags. These classification of tags as a whole for the entire document is always important. For example, to know amount of free text paragraph tag can be used. Below is a software that helps to classify the tags, This software is an open source in the Google web browser. The link to tag tool is `http://redwriteblue.com/tags/htmlcount.html`

The 19 test input examples have huge variation in terms of tag count, tag count is sum of all tags to better understand see below example:

Sum of all the Tags: 21+1+3+13+3+3+1+1+4+14+1+3+8+1+4=81

Test Input Name	**Classification of tags**		
	Tag name	***Open***	***Close***
Art Gallery (Id 1)	a	21	23
	body	1	1
	br	3	3
	div	13	13
	h1	3	3
	h2	3	3
	head	1	1
	html	1	1
	img	4	4
	li	14	14
	link	1	1
	meta	3	3
	p	8	8
	title	1	1
	ul	4	4

Table 5.1: When the HTML Test input is substituted in the HTML tag count tool the classification the tool performs on the input tags is illustrated

As we discussed there are many different types of tags present and they reflect different properties that are useful in building the web page applications. When the test input is given to the software it calculates the number of tags as a whole that are available in the file and represent them in the form of a table. Example ID and classification of different tags are addressed in table 5.1.

To answer the RQ.3a yes, we found some possible code metrics like number of tags and Depth of the node, Size and compress size. All the four metric together in a single table with Id of the example and the corresponding metrics are addressed below:

Test input	**ID number**	**Size**	**Compress size**	**Number of tags**	**Depth**
Art gallery1	ID1	5,310	2072	81	4
Art Gallery 2	ID2	5,310	2072	81	4
aerial1	ID3	1,867	918	34	4
aerial2	ID4	1905	931	34	4
Black Coffee	ID5	5921	1695	124	8
Lady tulip	ID6	5775	2323	114	4
Blue Media 1	ID7	10712	2451	143	6
Blue Media 2	ID8	9856	2247	143	5
Blue Simple template	ID9	24736	3392	253	13
Cooperation	ID10	8532	2576	158	5
Escape Velocity1	ID11	10186	2513	200	9
Escape Velocity2	ID12	10186	2513	200	9
Forty	ID13	7,485	2,012	149	7
Intensify 2	ID14	4,464	1,647	99	3
Studio1	ID15	14,654	3,386	239	7
Studio2	ID16	14654	3386	239	7
Coefficent1	ID17	5024	2237	102	4
Coeffiicent2	ID18	5024	2237	102	4
Intensify1	ID19	4464	1,647	99	3

Table 5.2: The Test inputs after the mutations are performed for all the Four metrics the following data is gathered for each test input.

5.2 Matching metrics from literature with test inputs

The existing metrics is size and compress size for that we can do number of character bytes in HTML, as they can be applied to any programming type we have chosen these general metrics for our study. We found metrics like depth of the nodes/elements in the source code since HTML is a tree like structure we would like to select the depth as a metric for this study. The HTML consist of different tags applied in it so we selected tags as a metric, both depth and tags are specific to HTML.

5.3 Preparation for experiment

For an experiment to be successful every step that is chosen is very important the test inputs depend on amount of significant variation they show among all the metric properties throughout the set. The experiment preparation is a lengthy process Since our study has to come up with things like:

- HTML test inputs.
- Metrics to be applied on HTML test inputs.
- Tool that can be used.
- Output representation.
- The mutation that are performed on the test input.
- The class-room and presentation setting.

5.3.1 Real life Examples Versus Automatically generated examples

Our Initial idea was to use real-life examples and if possible to use the random examples instead if the real life examples are not suitable.

Automated: The automated generated test inputs involves randomly generated HTML inputs with different properties imbibed within itself. Randomly generated test data may not have indentation in the same format as the normal HTML that is designed using Human so it is challenging and harder to comprehend.
Manual: The automatically generated test data could generate the test inputs with only certain features like small input larger depth or large input shallow depth. All the metrics do not have significant variation over the selected set of randomly generated test inputs. So, realistic test inputs both efficient and easy to comprehend over the randomized test inputs.

5.3.2 Test input

To reduce the scope of the project we want to select metrics that are used in software industry within Web development and metrics that are specifically applicable for the test data generation. In our study each participant of the experiment is provided with different HTML code samples and some possible HTML outputs. The participants are asked to go through the code and match the HTML source code with the respective output. So, the input HTML code and the respective outputs forms the test input of the experiment. After comparing the web page output with HTML code sample the participant can select any one option among the choices: input match output, input do not match output and do not know the answer.

1. Why not conduct the research using object oriented programs as test input?
In our reported results we are trying to argue only in terms of considering HTML. So, obviously a question arises why not use other programming languages unlike HTML. The software that we can use for the Java as test input to check whether the output is correct or not is through the Java compiler medium only. Moreover, 80 percent of the work done is completely related to Object orientation. Whereas the HTML test input had very few amount of work related to metrics that evaluate comprehensibility so, we choose HTML over Java as an input.

2. Whether to include only HTML or else JavaScript and CSS along with HTML?
We initially thought through to include JavaScript and CSS but then if they are included the validation of metrics throughout the code should be separately done. So, we exclude them and involve only HTML.

- In terms of testing, depth in CSS may refer something else when compared to depth in HTML.
- In terms of JavaScript, depth of tags in HTML is different from the depth of indentation of scripts, both the depths are valid but it depends on whether we are

including the JavaScript in HTML.

Moving on, as soon as they answer the output they can move to the next question, they cannot skip the questions. The output used in the Pilot study 1 is static output but based on the feedback from the pilot study 1 we changed the way the output should be represented. Instead of only static output images with no interaction for mouse over actions we also provided a folder with browser/ web page images which have interactive interface for the participants to easily answer the test input.

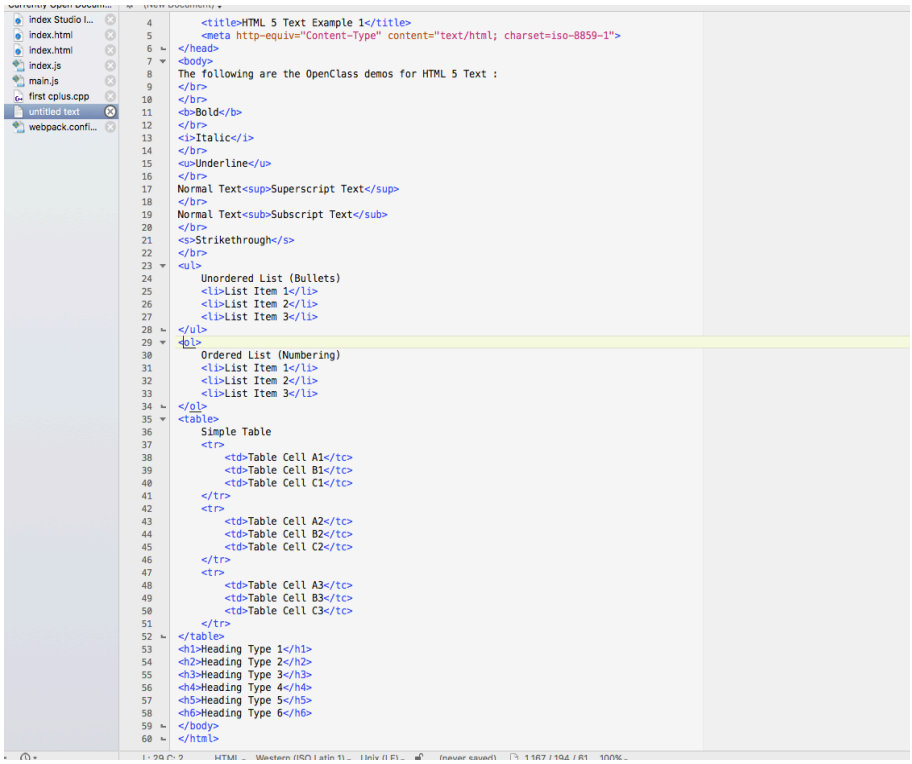
5.3.2.1 Selection of Test Input examples

We need to select inputs that has features and shared variation among the four metrics which are reported from the literature. The amount of data that is freely accessible is millions of lines of code so we primarily chosen the trust worthy websites like Git Hub and source forge. From these websites a subset of test inputs are needed to be selected for the project [62] [93]. Nevertheless, there are many important key points to discuss about the test inputs.

- First Set:
  - This first set is from the Git Hub, this Repository consists of 101 examples that are designed using the HTML CSS and JavaScript. From the selected 101 there are two important problems that we noticed.
  - Firstly, the HTML test input has JavaScript mixed with HTML and most of them are duplicates. If this is the case, then checking the entire HTML test input is a harder task.
  - Secondly, when depth of these test input is calculated, not so much variation is witnessed in it and those which vary significantly are not satisfying other properties like size and compress size so it hard to stick to these examples.

  First set test inputs are small and crisp but they are not suitable, as they do not show significant variation among metrics. Moving on, we started our search for second set and its advantages over the first set is explained as follows.

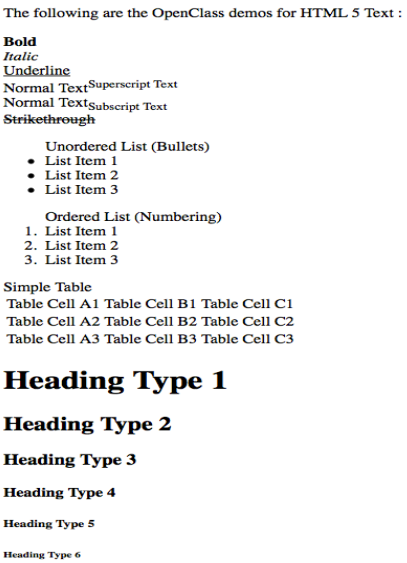
First Sample Set
First HTML Example Input and its corresponding output:



```
<title>HTML 5 Text Example 1</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>
<body>
The following are the OpenClass demos for HTML 5 Text :
</br>
</br>
<b>Bold</b>
</br>
<i>Italic</i>
</br>
<u>Underline</u>
</br>
Normal Text<sup>Superscript Text</sup>
</br>
Normal Text<sub>Subscript Text</sub>
</br>
<s>Strikethrough</s>
</br>
<ul>
    Unordered List (Bullets)
    <li>List Item 1</li>
    <li>List Item 2</li>
    <li>List Item 3</li>
</ul>
<ol>
    Ordered List (Numbering)
    <li>List Item 1</li>
    <li>List Item 2</li>
    <li>List Item 3</li>
</ol>
<table>
    Simple Table
    <tr>
        <td>Table Cell A1</td>
        <td>Table Cell B1</td>
        <td>Table Cell C1</td>
    </tr>
    <tr>
        <td>Table Cell A2</td>
        <td>Table Cell B2</td>
        <td>Table Cell C2</td>
    </tr>
    <tr>
        <td>Table Cell A3</td>
        <td>Table Cell B3</td>
        <td>Table Cell C3</td>
    </tr>
</table>
<h1>Heading Type 1</h1>
<h2>Heading Type 2</h2>
<h3>Heading Type 3</h3>
<h4>Heading Type 4</h4>
<h5>Heading Type 5</h5>
<h6>Heading Type 6</h6>
</body>
</html>
```

Figure 5.4: The first sample test input, IDE applied here is Text Wrangler.

Output of Sample 1:



The following are the OpenClass demos for HTML 5 Text :

Bold
Italic
Underline
Normal Text Superscript Text
Normal Text Subscript Text
~~Strikethrough~~

Unordered List (Bullets)
- List Item 1
- List Item 2
- List Item 3

Ordered List (Numbering)
1. List Item 1
2. List Item 2
3. List Item 3

Simple Table

Table Cell A1	Table Cell B1	Table Cell C1
Table Cell A2	Table Cell B2	Table Cell C2
Table Cell A3	Table Cell B3	Table Cell C3

Heading Type 1
Heading Type 2
Heading Type 3
Heading Type 4
Heading Type 5
Heading Type 6

Figure 5.5: The first sample test output, Browser used here is Google Chrome.

The mentioned example is not significantly promising to explain a good scenario, but two reasons to avoid the first set are metrics does not show significant variation on the test input and the output as well is not promising and takes lesser time to solve them.

- Second Set:

- We concluded to a set of 12 test inputs as better once over the Previous set. The outputs are much more responsive. The CSS and JavaScript are included in separate folders from that of the HTML Test input.
- Some Test inputs have mouse click actions, such complex multiple pages interaction makes the Test input hard to get it correct in the given amount of time. The selected test input has only 1-page application with no mouse click actions. Our study does not aim at checking the knowledge level of the participant.
- Mouse over actions only help to access links but they do not have redirecting functionality to new web page this can be done using mouse click actions.
- Selection of test inputs which does not involve the dynamic JavaScript and other libraries' within is very important. If it involves dynamic code library functions and frameworks, then the metrics are not going to be good representation of what people are going to measure.

The second set of test inputs have clear indentation, significant metrics variation and is Systematic they are significantly more promising than the previous 101 set. After comparing figure 5.1, 5.2,5.3 and 5.4 we concluded that the second test set is better over thee first set for performing mutations and experiment.

Sample Set 2
Then changed the HTML test input choices and shifted to examples that are more interactive:

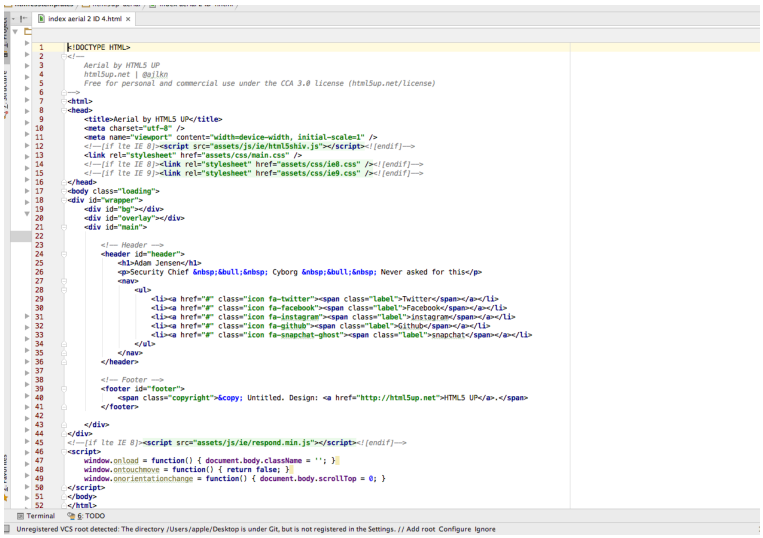


Figure 5.6: The second sample test input, IDE applied here is Text Wrangler.

Output of sample 2:
The output of the above given test input:

Figure 5.7: The second sample test output, Browser applied here is Google Chrome.

If we see carefully from the above two output images figure 5.2 and 5.4, then the resolution screen that is being completely utilized to display a more responsive output is varying. The second case 5.4 is most likely better interactive, responsive to the participant. A template if it is more responsive then preforming mutations on the template is more feasible and in case of HTML it is very important [94].

5.3.3 Selection of Tool for the Experiment

After gathering the right test inputs for the study then the tool which displays the input and output should be considered. There are some important criteria needed to be met before considering the tools is an apt once for the experiment.

- The tools should be able to display the outputs in the form of images.
- The tools should be able to calculate time taken to answer each and individual question by the participant.
- The tool should be able to display both input and output, the test input is in mange format because the test input cannot be modified or copied as if they copy the test input and paste them in the browser they get access to actual output itself.
- The tools should be able to perform randomization automatically.

Experiments with humans are always time consuming and we have to make sure that the tool we provide for them is easy to interact and watch for test input and get the output correct. To identify the right tool, we have gone through several available options that can meet the above requirements. We would like to discuss on the tools that we checked for the validation of requirements and see if they match up or not. From table 5.1 the lime survey is satisfying all the requirements for this study.

Tool Vs requirement	Time Stamping	Test input image display	Cannot Copy test input	Email invitation	Randomization
Survey Monkey	No	No	Yes	Yes	No
Lime Survey	Yes	Yes	Yes	Yes	Yes
Question Pro	No	No	Yes	Yes	Yes
Excel	No	Yes	Yes	Yes	No
PDF	No	No	No	No	No
Google Forms	No	No	Yes	Yes	Yes

Table 5.3: Different Survey tools that can be applied for the study and do they match the requirements of this study are illustrated.

5.3.4 Randomizing the question

Randomizing the test inputs is very important for our study in fact it is as important as the time stamping for individual question. The reason for randomization is recording the time for each and every question attempted by the participant is important so the questions given to each person should be of different order [95] [96]. We could perform manual randomization but sometimes even we miss out certain patterns that will show impact on the participants answering ability. We performed randomization using the tool Lime survey which has the ability to perform randomization automatically. We gave every question an identification number ID number this enables us in which pattern the questions are being answered by the participant.

5.3.5 Class Room Setting for the experiment

The Class room setting especially when the research method that is applied is an experiment will show high impact on the participant [97] [98]. For a controlled experiment to get it correct all the external impacts on the experiment like all the participant are given the test inputs and they are asked to evaluate the outputs. The systems should be having common interface they have a good reasonable amount of Internet speed, they all work on same resolution settings,all are compatible to run with Lime survey, they all are having the browser compatibility with Google Chrome and speed of all the systems are at same level.
Presentation section: The presentation section which is a demonstration given to the participant before is the start of the experiment and it is very important because what is being told to the participant affects the way they evaluate the test inputs. If anything that is related to evaluation of metrics are directly told they do impact the way they evaluate the test input.

5.3.6 Mutations on test inputs

Mutation is performing small changes in the code yet it gets complied and the output is displayed, mutations constitutes for modification of programs. These program modifications are done in small scale. High amount of research is being done to apply mutation analysis within the non-procedural and object oriented languages [92].

Traditional mutation analysis is a code based method which invokes the ability to apply small sensitive syntactic changes and these changes are performed on the structure of

the program [99]. When a single change exactly once is applied on the program using some mutation operators then a single mutation program or simply mutant is produced [100]. The equivalent mutants are those which have same input or output relation as that of original program [32] [33]. So we need to consider equivalent mutations which is changing the program yet the outputs are still operational in the same way.

We performed sensible mutants on the test inputs. In case of a program we can apply the mutation operators. However, We cannot apply mutation operators like + with * and – with / these are applicable to object oriented programs unlike in HTML these operators will not show significant amount of visual impact on outputs so we have to consider other ways to perform mutations [35]. when displaying the image as an output the performed mutations should be noticeable to the participant so that the they can identify if it is correct output or not. If there are no mutations that can be performed on the test input, we must almost come up with new mutations that has a visual impact. followed some steps which are as follows:

- For the original Test inputs that we have which is a count of 12 test inputs We considered them as the original test inputs It is given as P1.
- Then from the original test inputs 12 P1 we created 2 duplicates for these original test inputs. The duplicate File names are HTML 1 and HTML2.
- These 2 duplicates have each with one mutation performed on the original test input P1.

As said earlier the initial set of 12 test inputs are worked out and we performed mutations on these test inputs in the files HTML1 and HTML2 duplicates without manipulating the original P1 file set. Some important criteria for mutation are as follows:

- The changes performed on the test input should not be too small for example removing the white spaces doesn't not show significant impact on the output that is being tested. Similarly, should not be a large mutation change like alternating the images (Changing the color of the background or image).
- The number of mutation performed on the test input also impacts the time taken to get them correct.

Mutation Score table is given above in the table 5.2

As our test inputs are HTML and the output is a browser that we are testing therefore So, we fixed to only single output. The single output forces the participant to look at the entire test input which is usually how it should be done to look at the entire code and see if it matches with output which sustains the justification towards correct way to comprehend the test input.

S. no	Test Input	ID	Which file is modified	Output	Output displayed	What are the mutations performed on test input
1	Art gallery 1	ID1	HTML1	Wrong Output	Original output	1) Paragraph 2 and 3 under side heading welcome to our website are interchanged 2) Both the paragraphs are interchanged in aliquiam section. 3) Quick links and portfolio links interchanged.
2	Art gallery 2	ID2	HTML2	Wrong Output	Original output	Quick links and portfolio links are interchanged.
3	Aerial 1	ID3	HTML2	Wrong Output	Original output	1) The lines under Adam Jensen is changed by removing full stop in between. 2) The icon Dribble in original file is replaced with Instagram.
4	Aerial 2	ID4	HTML1	Wrong Output	Original output	The icon Dribble in original is replaced with Instagram.
5	Black coffee	ID5	Original	Wrong Output	HTML2 output	1) Increased the paragraph from original size. 2) Interchanged the paragraph
6	Lady tulip	ID6	Original	Correct Output	Original output	No changes are performed on the HTML test input
7	Blue media 1	ID7	HTML2	Wrong Output	HTML2	1) Category are interchanged which is in the form of link and highlighted. 2) Image files are interchanged.
8	Blue media 2	ID8	HTML1	Wrong Output	HTML1	The time is replaced in both the sections.
9	Blue simple template	ID9	Original	Wrong Output	Original output	1) Headline and slogan text are interchanged. 2) The table at the bottom is highlighted.

10	Cooperation	ID10	HTML1	Wrong Output	HTML1	Text area is replaced by text field.
11	Escape velocity 1	ID11	HTML1	Correct Output	HTML1	No changes are performed on the HTML test input.
12	Escape velocity 2	ID12	HTML2	Wrong Output	HTML2	Buttons are replaced with different color.
13	Forty	ID13	Original	Wrong Output	Original output	1) Message field is interchanged with phone field. 2) Text under the aliquimis replaced with different text.
14	Intensify 2	ID14	HTML2	Wrong Output	HTML2	1) Headings are interchanged. 2) Feugiat lorem is replaced to Ferrari lorry. 3) Phone number at the bottom is changed.
15	Studio 1	ID15	HTML2	Correct Output	HTML2	No changes are performed on the html test input.
16	Studio 2	ID16	Original	Wrong Output	Original output	Changed the images.
17	Coefficient 1	ID17	HTML1	Wrong Output	HTML1	1) Changed the log instagram to twitter. 2) Interchanged text marius luctus and Maecenas vulpate.
18	Coefficient 2	ID18	HTML1	Correct Output	HTML1	No changes are performed on the HTML test input.
19	Intensify 1	ID19	HTML2	Correct Output	HTML2	No changes are performed on the HTML test input.

Table 5.4: The Test inputs selected for the entire study, the mutations performed on each test inputs are clearly illustrated.

5.3.7 Representation of output

Displaying output through browsers is advancing day by day, they can incorporate the functionality of already existing browser features and also more sophisticated features can also be displayed [101]. These web based search engines display all types of content to the client and they can even interact with graphic interface in a more flexible manner [101].

Weber [102] uses browser as an interactive medium to display the HTML and XHTML inputs.

Initially for our study we thought of several alternatives for displaying the outputs these among these alternatives we concluded to choose single image display. The alternative options that are thought through in the process are addressed below:

- Participants able to draw the outputs. This is very hard task if the given test input scenario is complex.
- To print the outputs and present to the participants. This impacts the time calculation as time stamp is important for every question and must be calculated it is avoided using the paper as a medium.
- To validate one single output and see if it is correct or not. This is the representation choice we adopted over the others as it allows the participants to go through the test input step by step to identify if it is correct or not.
- To give the participants multiple outputs. Each output shall have either one mutant or more than one mutant performed and they have to identify which one is matching the test input. The problem in this case is they tend to identify the difference among the outputs displayed if the difference is found they certainly look for only that particular section of test input to select the correct choice so this choice is avoided.

If we allow them to interact with the actual website, they interact with the HTML and compare them with the source code using the developer tools. It is best to avoid such type of displaying of direct websites.

5.4 Pilot Study and Experiment

The results from the Pilot Studies and experiment are in the table format and as they occupy more number of pages so they are included in the appendix E section. Here we have presented the process that is involved in the pilot studies and experiment.

5.4.1 Importance of Pilot Studies before conducting Experiments

The literature on pilot study is considerably less but the studies from Thabane et al. [103] describe they contribute significantly towards improving the study. How to conduct the pilot studies and who to choose as the participants and steps in pilot study are addressed Thabane et al. [103]. Participants that are chosen for the pilot study should be capable enough and selected based on objectivity but not on the basis of recommendations [104]. R.L. Glass [104] describes the steps to be followed while implementing pilots which are namely as follows

- Pilot planning: planning such that Pilot to be conducted is linked to problem under study.
- Pilot design: defining conduct, execution, identify the data to be gathered, from where the data is drawn from.

- Pilot conduct: conducing pilot by following the design made.
- Pilot execution: recording problems and draw conclusions.
- Pilot use: changing the implementation decision based on the analysis conclusions.

Leon et al. [105] relates pilot study to success of the research project. The pilot studies term is frequently used in the research reports, the contribution for research made by the pilot studies are not always explicit [104]. The pilot studies don't help in validating the hypothesis rather it acts as an early study that enhance the probability of upcoming experiment [105].

It is important to conduct the pilot study in our research design as it often helps to determine the size of the test inputs, time given to the participates, Information on target population and other factors that are taken into account are sufficient [106]. The pilot population should be quite similar to that of target population otherwise it is meaningless [106].

5.4.2 Design and Use of Pilot Studies

The pilot study is very important for this study because the test inputs that we are giving to the participants might have some challenges like we don't know how much size of the test input constitutes. Whether the test inputs that we have selected are good enough to answer, are there any necessary changes that are needed to be done on the test input. Are the participants able to solve the test inputs within the time allocated all these questions can be answered by pilot study.

5.4.2.1 Pilot Study 1

Pilot study main agenda is to conduct a workable environment which replicates in a same way as the experiment is being conducted. The participants in the pilot study are known colleagues whom we have consulted for their assistance to participate in the pilot study and give us the feedback. The selected test inputs for the Pilot study 1 and their corresponding ID's and all the four metrics variation are illustrated in appendix E-1. The analyzed data for the pilot study 1 is presented in the Appendix E section. The participants don't need to know about what metrics are induced into the test input. They are given the test input and output in the form of browser,they both are compared to see if the match or not.

During presentation: It is important to convey correctly, also how much information does the participant need to know should be constrained and protected. Any chance in revealing more information about the idea and estimations of the project would impact the way the participants look at the test input. The presentation announcement is addressed in the appendix C.4 section. The questions that are taken for the pilot study 1 are represented with their id numbers in table 5.5.

Pre-Questionnaire:

The pre questionnaire includes Very important questions like their knowledge in HTML and the expertise level in HTML. The data that is gathered in pilot study one is addressed below:

- Number of participants attended are 4
- Out of 4 participants 2 of them have intermediate level of knowledge in HTML and remaining two of them one is an expert and the other remaining participant is having basic knowledge in HTML.

The participants are then requested to attend the Pilot study one which is conducted in the lab at BTH university.

Use of the Pilot Study 1

The feedback is given by the participants from the questions that are given to them using the Google forms as a medium. Now the question that are asked for the participants are addressed in the appendix section. Feedback given by the 4 participants in this pilot study helped to improve for the final experiment.

Avoid Likert Scale: During the experiment the participants selects the multiple choice and also address the corresponding Likert scale however, this entire process is confusing and makes the participants work more difficult by giving ambiguous meaning. From the feedback we decided that the confidence level is not highly helpful for analysis and there is no need for including the confidence level so, after careful evaluation we avoided using the confidence level. This measure is reflected in our second Pilot study thus pilot study 2 is improved by avoiding the Likert scale. This is the first time we are conducting this kind of study so there are some challenges that we faced like all the systems did not function simultaneously due to server problems and lack of proper Internet connection. So, this problem is reflected in the participant's feedback. Thus, for the next pilot study we made sure that such challenges are mitigated.

Replacing static images with web pages: Out of 4 participants all 4 of them gave feedback that the static images are not helpful in selecting the multiple choice as the test input have several interactive features that cannot be deduced from a static image unlike in the case of browser link where they can access the output and validate mouse over actions and several other simple features that are hard to analyze from the static images. The images are displayed at the end so this makes them hard to compare the test input (Minimum number of lines 70 and sometimes maximum up to 350) with the output by always scrolling down this decrease their efficiency. As the number of lines increase the participants find it difficult to answer the test inputs. The format in which the experiment is carried out is first the test input image is displayed. Then three multiple choice questions to answer, then later is the static image. Now as the test inputs have greater than 70 lines of code it was hard for participants to compare by scrolling up and down.

The Participants mentioned they faced problems in experiment as the test input source code has some sections where they need to interact with the website to test if the functionality is working or not, this is observed in case of Text fields,buttons,hyper link and other libraries, so they requested to give web pages itself. After giving them the web pages it was easy for them to interact with the single web page and check whether all the hyper links are accessible. We are also convinced to give them the web pages based on their need as it is effecting their answering ability, in turn they are asking a lot of questions about the output image. Thus we have decided to given them one Test input and one

test output, the test input is image format cannot be copied and the test output is a web page in the experiment, web page given to them is a single page applications. But to avoid cheating and looking into original source code on line we managed to monitor them constantly so that they do not see actual code.

This particular study is very fortunate enough to get feedback on some of the key challenges that helped a lot in improving in further studies. Some participants gave feedback about the time constraints however, giving the participants 1 hour of time and also 10 questions are sufficient enough when all the above challenges like are satisfied.

- Alternatives to static image display and more clear pictures/image outputs.
- Removing confidence level for multiple choice options.
- Improved experimental setup.

So, for this study we decided to conduct another pilot study 2 with the above challenges met.

5.4.2.2 Pilot Study 2

The main purpose of conducting pilot study 2 is to mitigate the challenges faced in the first pilot study 1 and be prepared for final experiment. The pilot study 2 preparation involved selecting the test inputs that are not same as that of previous pilot study. Two out of four participants are requested to reappear in the second pilot study. For the Pilot study 2 which is conducted at The BTH university participants who attempted pre questionnaire all of them have attempted pilot study 2. the analyzed data for the pilot study 2 is presented in the Appendix E section.

1. What the participants are given?
   The participants are given the test inputs and outputs using the lime survey tool. Along with the static image outputs the live web page images are also given access to the participants. This live web images are given access using the Google drive. Just before the experiment starts the participants are requested to download the shared folder in the Google drive and extract the 10 outputs.

During Presentation: All the participants have been given similar instructions from pilot study 1 except the confidence level Likert scale point is not mentioned as this time in the pilot study 2 section the Likert scale is completely removed. The selected test inputs for the Pilot study 2 and their corresponding ID's and all the four metrics variation are illustrated in appendix E-2.

Pre-Questionnaire:

The pre questionnaire questions include participant's information like name, email, which group are they from, whether they have knowledge in HTML and what is their knowledge level in HTML are asked to avoid inexperienced people.

- 4 participants attended the pilot study 2, 2 out of 4 participants are reappearing for this study.

- 3 participants have intermediate level of expertise and 1 participant have expert knowledge level in HTML programming

Use of the Pilot Study 2:

In the pilot study 2 the outputs are displayed in the form of live web page links furthermore the confidence levels are removed from the pilot study thus the participants can easily select only the multiple choice options without any confusion. The feedback like giving information about the changes made on the HTML test inputs can should be addressed before is one feedback given by the 2 participants. To know what are the changes made to the HTML precaution is taken for next Experiment in such a way that the participants are informed prior that the changes are made in large scale as described in the mutations table this information about small changes are not performed on the HTML test input is conveyed before the experiment begins in the presentation section itself to the participants. Remaining feedback seems that participants are satisfied with the way in which the process is conducted and the entire setup seems improved from the previous once. At this moment it seems we are ready to work on the experiment and start conducting it to analyze how the data influence the metrics to understand which metric Is good predictor of human oracle costs.

5.4.3 Experiment Design and Execution

After conducting the pilot study 2, based on the data gathered and feedback taken from the participants we are confident enough to conduct and explore the real experiment.

Planning and designing Experiment

The experiment has same questions with test inputs taken from pilot study 1. The number of participants attempting the pilot study and the experiment is the only criteria that changes and moreover the improvements made in both the pilot studies are included in experiment.

Before the participants are invited to the experiment we made sure to send the pre questionnaires' to know their knowledge of HTML. The participants are sent a cover letter which includes conditions for attempting the experiment and also a Google form registration with different time slots in convenience with the participants. Both the cover letter and the questions asked in the pre questionnaire are included in the appendix section.

Pre-questionnaire

The pre questionnaire is sent one day before the start of the experiment. The pre questionnaire is important for our study as it helps to gather information about the participants like their name, mail id, which specialization are they from and also two more very important questions like their knowledge in HTML and the expertise level in HTML. We conducted multiple experiments due to participant's convenience, availability of lab and no technical interventions. The data that is gathered in experiment is addressed below:

- The cover letter and the pre questioner sent to the department of software engineering and primarily sent to only students who registered in master thesis. Not many of them appeared for this section so we had to re do the experiment as the participants are not sufficient.
- For the second section the pre questionnaire and the cover letter is once again sent to the participants who are registered for master thesis in the department of software engineering. Along with this invitation another invitation is sent to People from vinnova program initiated by BTH which primarily focus capability building program who are working in Soft house AB.
  - For the second invitation we gave 2 weeks' gap before conducting the experiment, gap is important because unlike last case more people can attempt.
  - Four sections are created for the participants to choose among them so that they can attempt the experiment based on their availability.
  - For the final experiment a total of 32 participants have appeared for the experiment and all the experiment are conducted in the same lab in the H block in BTH university. Out of 32 participants all the participants have experience in HTML.

Conduct the Experiment

The total number of questions given are 10 and they are instructed with the rules to know before the experiment.

The experiment is conducted in the following step by step procedure:

- All the participants enter the lab and take their positions to start the experiment.
- We welcome all the participants and make sure to wait 5 to 10 minutes so that all participants arrive.
- The participants are instructed to login using their university acronym and password.
- Then after the participant's login they are instructed to open the Google drive folder and download the file shared to them.
- The participants are instructed to only open the index.html file using the Google chrome browser and we as examiners monitor their work and see everything is going as planned.
- Each participant has access to 10 outputs which match the number of question which is 10 and we made sure that all participants have the access and are functioning properly.
- A brief description presentation is given about experiment.
- The participants are sent the link to their mail id's, this link comprises the registrations for the experiment. All the questions are in randomized order.

- After 1 hour the section automatically stops and the participants are instructed to wait a few moments and the post questionnaires is sent to the participants.
- Then after the post questionnaires the participants are informed about the group interview section and instructed about the group interview.

Execution of the Experiment

The question id for test inputs represents which test input questions are used for this experiment, the experiment has same questions that are used for the pilot study 1.

Results from Experiment: In the experiment, which participant answers to the question is not revealed as a matter of privacy violations. Thus along with the existing 4 metrics from the literature a new set of four metrics from the feedback and group interviews given by the participants are obtained all these together represented in one single table which is addressed in the appendix E section.

5.5 Results from Group Interview

Procedure applied: As soon as the participants finalize the post questionnaire they are requested to stay back for a moment and informed about the group interview. Conducting the group interview right after the post questionnaire is finished helps to instantly gather the feedback from the participants. The questions asked to the participants are as follows:

1. Do you think given HTML test cases have any difficulties to get them correct?
2. Any of these difficulties that are specific to HTML test input?
3. Why were the test cases difficult?
4. Do you think the test cases are different or some of them more difficult than others?

The group interview time span is 15 minutes. All the interviews are transcribed and stored to perform further analysis based on the feedback and knowledge the participants shared. Four experiments are conducted out of which three experiments are include with the group interview as part of the experiment protocol. For one experiment we were unable to conduct as we did not design the interview. However, between 1st and remaining three experiments there is 2 weeks to design and get ready with the protocol. Some important conclusions that are drawn from the feedback given by the participants are as follows:

Interview 1: Total 4 participants

- Participants mainly focused on looking for tags and indentations whether they are correctly represented or not.
- Participants support that number of tags present does impact the HTML test inputs.
- Links in the experiment if they are working or not and does the web page show mouse over actions for the corresponding links given in HTML.

- Length of code, 2 out of 4 participants think that as the number of lines of code increases the time taken to answer vary.

Interview 2: Total 7 participants

- 1 participant say it is not that hard to figure out the output whether it is correct or not.
- 1 participant suggested the length of the code impacts the reader in answering the questions which is again supported by other 5 participants.
- 1 participant mentioned Time factor is significant while answering the questions it influences the understating of the code.
- 1 participant suggested the position of the mutant applied on the test input might the concentration level and brings negligence into picture while answering the test inputs.

Interview 3: total 18 participants

- 1 participant mentioned when the code length is very long it impacts the test input this argument is supported by 10 participants.
- If the test input has java script and other languages included in them then it is easy to compare the colors and additional classes and work through the code much easily this argument is raised by 5 participants. However, not all know the CSS and JavaScript but after a careful discussion which went through among the participants 3 people supported this argument that including the CSS and JavaScript might reduce and impact number of participants attending the experiment as not all have good expertise in CSS and JavaScript.
- 1 participant suggested it takes more time than now to answer the questions when CSS and Java Script is included and this argument is supported by 9 participants.
- 1 participant strongly mentioned that the output and inputs should be clear and understandable to see and validate how much of the test input is matching the output GUI and the test inputs should match each other in terms of colors used, the id's and selectors should have the CSS for more clear evaluation of output GUI
- 1 participant with industrial experience mentioned the color combinations and font is nice and the tools used in the industry are more sophisticated and have advanced testing feature like collapsing the div tags using the IDE's and look for the part which you want to compare and move to next section. when the company gives the participants raw code/test input the IDE's does most of the work for the test. IDE's does half of the work in real time testing and the comments are very clearly mentioned to understand and go through the test input.
- 2 participants argue that in some case the color of the font directly matches with the background.
- 1 participant mentioned amount of text that is being used into the paragraphs impact the readability as the time taken to check the entire paragraph with the output is challenging. This argument is supported by 12 participants.

Selection of right metric based on conclusion drawn from the feedback given by the participant.

a. Firstly, instead of counting number of links that are present in the test inputs, the anchor tag which helps to include the links in the HTML test inputs are taken into account, so the anchor tag can be useful as a metric.

b. The lines of code are a different size metric from previous version that is used already so we decided to include both. All the HTML test inputs does not have any CSS and JavaScript so lines of code for the entire test input is taken into account.

c. The tag in the HTML indicates different sections in the web page. As the sections increases the time to check the input with output increases. So, the <Div> tag is taken as a metric.

d. The amount of free text is usually addressed in the paragraph <p> tag so we included this as a metric to understand if this metric is a good predictor of human oracle costs.

Question Do participants perceive of any new metrics that might influence the correctness of the test input?

Motivation: To answer this research question yes, from the group interview, we believe there are some new metrics that are to be taken into account in this study. These conclusions are drawn from the feedback and transcribed interviews. The following are the new metrics that might influence the correctness of the test input with output.

Question ID	**LOC**	**Div**	**anchor**	**<p>**
1	133	13	21	8
3	52	4	6	1
6	134	29	26	11
7	210	47	33	8
9	383	63	59	8
11	300	46	36	19
13	217	13	23	8
16	370	92	16	35
17	120	27	22	10
19	152	22	12	8

Table 5.5: The Metrics drawn from the interview questions asked to the participants as part of the experiment.

Chapter 6

Analysis of the results

The literature review results help to answer the research questions RQ1 and RQ2, while paving the way for research question RQ.3a (The RQ3a answers some possible code metric that can e applied on test data) then later the RQ.3b is obtained after the Experiment is conducted (The RQ3b answers which among the selected metrics show influence on human time and accuracy). While the chapter 4 focuses on presenting the setup and results obtained from the experiment. The analysis of the experiment results is addressed in this chapter 6.

6.1 Regression Analysis:

D. E. Berger [107] articulates that for estimating technical efficiently the regression analysis can be employed. The regression analysis can be useful to find relationships between multiple number of inputs and outputs [107]. For the comparative efficiency the Regression analysis and data envelopment analysis are useful [108]. E. Alexopoulos and Liang et al. [109] [110] says the regression analysis is described as methodology that helps to identify functional relationships between two or more variables. This is represented in the mathematical form which helps to predict the value of one variable from value of another variable [107] [111] [112]. The regression equation is defined as [95] [113].

$$Y = \alpha + B_1X_1 + B_2X_2 + B_3X_3 + \epsilon. \tag{6.1}$$

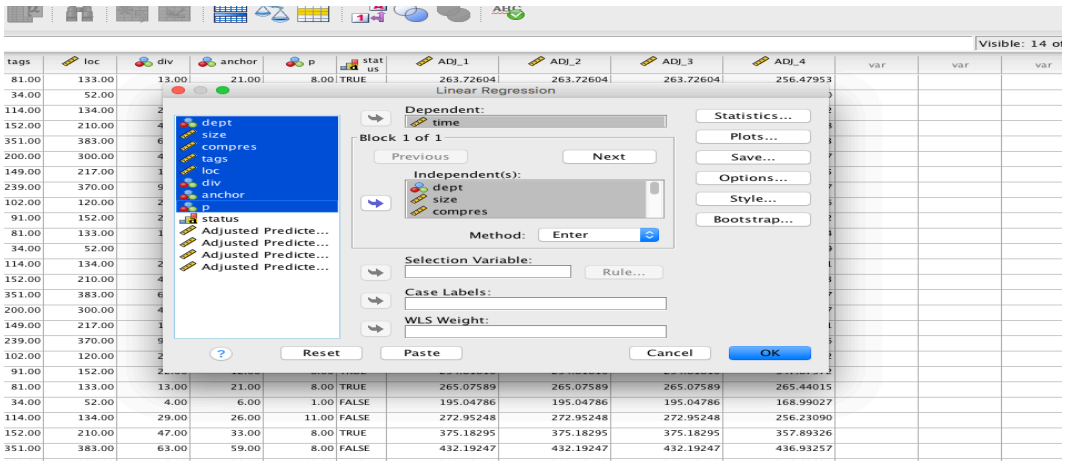


Figure 6.1: SPSS statistical tool that helps to perform the Regression analysis using dependent and independent variables are illustrated.

The regression analysis helpful to find various factors lie correlations, significance, tolerance, P-p plots, Variation indicator factor VIF, R square value [114] [115] [116] [117]. In the book the author explains how to implement SPSS in 23 steps [115] [118]. The figure 6.1 and 6.2 addresses how to use SPSS to perform regression. If you have one dependent variable and more than one independent variable, then we apply multiple regression analysis [108] [119]. There are a lot of options in SPSS to perform statistics [115].

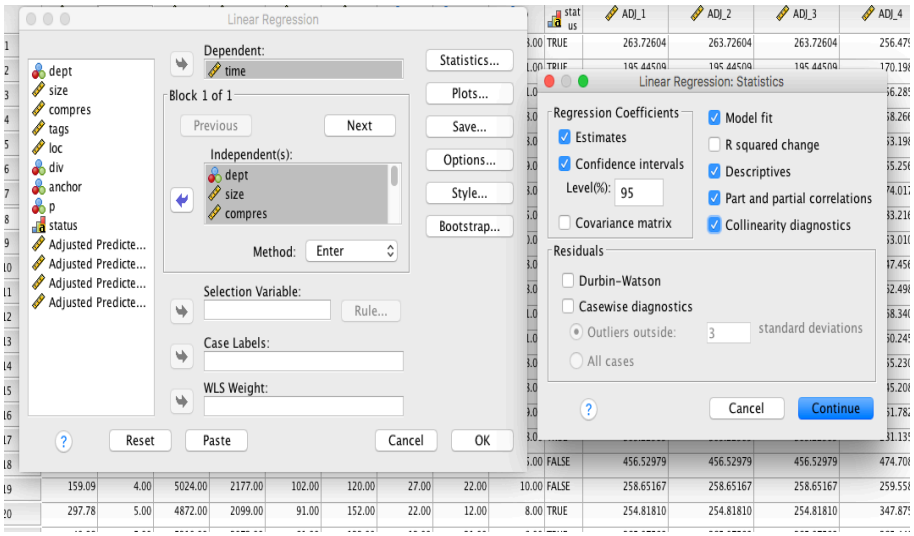


Figure 6.2: SPSS statistical tool helps to statistically calculate many different statistic's based on the convenience of the researcher.

6.2 Time dependent variable vs the metrics independent variables:

The Figure E.5 in appendix section contains the time taken by each participants to answer each question which include all the 32 participant's results. The variation in the metrics across all the test inputs are displayed. From the results obtained after performing the regression analysis on the data points.

6.2.1 Pearson Correlations among the Independent Variables

The relationship between independent and dependent variables can be identified from the correlations table. From the Pearson correlation column in the correlation table it is clear that all the independent variables have positive relationship with time this is noticed in first row across time in Pearson correlation column. Among these independent variables size is highly correlating with time (0.328) followed by lines of code (0.327) then (0.323) and at last anchor (0.187) is positively correlating but considerably small when compared to size. The correlation helps to understand how different variables interact, correlate with each other. If the values are greater than 0.7 the this is not good for the model in this case however all the values are less than 0.7 [111]. From the table if observed all the independent variable have positive relationship with the dependent variable time.

Correlations									
		time	***dept***	***size***	***compress size***	***tags***	***loc***	***div***	***anchor***
Pearson Correlation	***time***	1.000	.193	.328	.322	.302	.327	.323	.187
	dept	.193	1.000	.645	.549	.432	.655	.528	.162
	size	.328	.645	1.000	.933	.908	.967	.920	.639
	compress size	.322	.549	.933	1.000	.929	.950	.851	.710
	tags	.302	.432	.908	.929	1.000	.951	.799	.811
	loc	.327	.655	.967	.950	.951	1.000	.867	.651
	div	.323	.528	.920	.851	.799	.867	1.000	.436
	anchor	.187	.162	.639	.710	.811	.651	.436	1.000
	p	.241	.708	.684	.650	.455	.654	.808	-.005

Table 6.1: The correlations of all the 8 metric variables selected for the study, In this case all the metrics are positively correlating with time.

6.2.2 Linear Regression Model

From the below Model summary diagram if observed carefully there are R Square, Adjusted R square and standard error for the estimate. R square value is 0.126 that is 0.126*100= 12.6 % of the variance in the dependent variable is explained by the independent variables. That means how much of the variance in time is perceived by the independent variables is deduced from the Summary model table. For a good prediction the model should have enough variability or variance to find out the variance the regression analysis is helpful. One important assumption while performing regression analysis is the size of the data set used bigger the size better the regression helps to predict the outcomes [112]. The adjusted R square is similar to that of R square but rather it's more useful when the sample size is very small and when the sample size is big then the R square value should be considered.

Model Summary		
R Square	***Adjusted R Square***	***Std. Error of the Estimate***
.126	.103	238.70613

a. Predictors: (Constant), p, anchor, dept, tags, div, compress size, size, loc
b. Dependent Variable: time

Table 6.2: The Model Summary table illustrating primarily R value, R square values.

The coefficients table is the most interesting table as it helps to identify the relationship between the independent variables and the dependent variables. Larger beta value under the standard coefficients value column suggest that the predictor value have a large impact on the criterion variable. Similarly, a large t-value paired with small significance value suggest that the predictor value have a large impact on criterion variable.

In the coefficients diagram table 6.3 given below consider the lines of code lines of code the t-statistic is 0.828 and significance is 0.408 is p value not significant as it is greater than 0.05. If the entire significant columns are taken into consideration the values of depth 0.948; size p value =0.358; compress size p=0.390; tags p= 0.352; lines of code p=0.408; tag p=0.93; <anchor> tag p=.948 and for <p> tag p= 0.948. The p

value in all the cases are greater than 0.05 which means the model is not significantly the reason is the selected independent variable have high multicollinearity among each other.

From the table 6.3 if observed carefully it interprets the column standardized coefficients Beta, t and Sig p under the coefficients table diagram. Here standardized means the different independent variable that are used in the regression these independent variable values are converted to the same scale for easy comparison. The Beta value for lines of code is 0.937 this value is highest among all the existing variables which interprets of all the independent variable that are present the beta value of lines of code makes the largest contribution for predicting the outcome.

If the first column is observed carefully then we have model variable names and in second and third column which include B and standard error. Some of the independent variables for which the B values are negative like depth(-1.999); and for some of them the B value is positive like size= 0.131; the question is does these signs make any sense to answer that the coefficients on the independent variables in multiple regression for a 1 unit increase in the independent variable depth/level of HTML test input the model predicts the dependent variable time decrease by 1.999 units and all other independent variables are constant. The increase or decrease in the dependent variable depends on the sign of the value in B. The lower bound and the upper bound which indicates the values lies in between this interval and in case if the model is significant these 95 percent confidence level interval range should be very small almost nearer to zero. The multiple regression equation built from the coefficient table is as follows:

$$\begin{aligned} y = 52.464 + (\ \ 1.999)(Depth) + (\ \ .026)(Size) + .131(compresssize) + \\ (\ \ 2.283)(numberoftags) + 2.231(linesofcode) + 5.469(< div >) + \\ (\ \ .447)(< anchor >) + (\ \ 11.624)(< p >). \end{aligned} \tag{6.2}$$

Coefficients								
Model		***Unstandardized Coefficients***		***Standardized Coefficients***	***t***	***Sig.***	***95.0% Confidence Interval for B***	
		B	***Std. Error***	***Beta***			***Upper Bound***	***Upper Bound***
1	(Constant)	52.464	185.222		.283	.777	-311.983	416.910
	dept	-1.999	30.359	-.018	-.066	.948	-61.733	57.735
	size	-.026	.028	-.407	-.921	.358	-.080	.029
	compres	.131	.152	.370	.861	.390	-.168	.431
	tags	-2.283	2.450	-.791	-.932	.352	-7.104	2.539
	loc	2.231	2.694	.937	.828	.408	-3.069	7.532
	div	5.469	3.250	.554	1.683	.093	-.926	11.864
	anchor	-.447	6.821	-.025	-.065	.948	-13.869	12.975
	p	-11.624	9.481	-.409	-1.226	.221	-30.278	7.030

Table 6.3: The Coefficients table illustrating standardize and un standardized Beta values, t value and P(sig) value.

From the collinearity statistics in the coefficients table 6.4 below if observed carefully then it contains the tolerance and VIF variation inflation factor columns. If the VIF value is 1 then there is no multicollinearity among the variables. The tolerance indicates how much of the variability of that particular specified predictor variables is not explained by other variables in the model. The tolerance value is very small that values with tolerance<0.1 indicates high multicollinearity and the variables have correlations with each

other. The table diagram interprets that all the independent variables have tolerance value<0.1 which indicates they have high multicollinearity. For the depth the tolerance value is 0.039 which means 3 percent of the variance in the depth independent variable is not being accounted by the other independent variable. This value is obtained by dividing the VIF variation inflation factor 1/25.880=0.039. Similarly, the value of Variation indication factor VIF >10 which indicates there exists multicollinearity. This VIF is quite opposite of tolerance. The VIF which predicts the multicollinearity should be less than 10 VIF <10 for each variable. In the below case no variable that is independent variable have VIF <10 this is a concern so to solve this challenge it is addressed in dealing with multicollinearity section. Here the zero order, partial and part are not important to this study so our primary focus so they are avoided.

Coefficients						
Model		***Correlations***			***Collinearity Statistics***	
		Zero-order	Partial	Part	Tolerance	VIF
1	(Constant)					
	dept	.193	-.004	-.003	.039	25.880
	size	.328	-.052	-.049	.014	69.567
	compress size	.322	.049	.046	.015	65.587
	tags	.302	-.053	-.049	.004	256.654
	loc	.327	.047	.044	.002	455.385
	div	.323	.095	.089	.026	38.515
	anchor	.187	-.004	-.003	.019	51.751
	p	.241	-.069	-.065	.025	39.493

Table 6.4: The Coefficients table illustrating Collinearity statistics (Tolerance and VIF Variation Inflation Factor)

Appendix table E.1 table shows the regression analysis is performed separately for each individual metric, that is time dependent variable versus the metric independent variable. All the metrics show significance which is p<0.05 this data is concluded from the last column, all the metrics show such significance thus null hypothesis for individual metric vs time can be rejected in all the cases.

The appendix table E.2 shows the regression analysis is performed separately of combination of metrics and time that is time vs two metric combinations and the corresponding regression analysis gave interesting results, the * indicates the corresponding value is significant. For example, in the time vs depth and size, size has p<0.05 which means when two metrics depth and size are considered the variance in time is significant only in the case of size.

6.2.3 Conclusions and Challenges in the regression model

Conclusions: Firstly, from the model summary the R square value indicates there is 12.6% of variance in the dependent variable is indicated by the independent variable. The model is statistically significant with p<0.05, the independent variables have positive correlation with dependent variable and the size is highly correlating with time followed by Compress

size and div tag. The metrics have very high multicollinearity among themselves.

Challenges: Even though the model summary describes the 12.6 % of the variance in dependent variable is indicated by independent variable which specific one is showing such variance is hard to conclude. 12.6% is very low which means we cannot over claim from R Square. The low R square values clearly indicates there are some variables clearly missing which we haven't taken into account these values can be metrics of any form may be related to source code or even person.

There is multicollinearity in the coefficients table and if the multicollinearity exists, then the model is not significant. when we selected the examples the constraint was to avoid collinearity as far as possible and yes we selected test input with metrics showing significant variation but we were unable to avoid multi collinearity. For us multicollinearity is not a big surprise, it was always there from the beginning and the ideas is to pick the test data that minimizes the multi collinearity. To minimize the multicollinearity we have remove more collated independent variables step by step from the model which is what we did in the next section 6.2.4.

6.2.4 Reducing the Multicollinearity

Description about challenge: In this study both the independent variables and dependent variables are continuous variables. If both the independent variables and dependent variables are continuous then the regression is the better way to perform statistical analysis [120]. Firstly, it is important to understand what is continuous variables. Basically some common variables types are continuous variables categorical variables discrete variables there are other types of variables as well but for this study say these three are relatively sufficient [120] [121]. The categorical variables are those which can be classified into different types of categories like car colors, perfume brands and so on [122] [123]. The continuous variables are those which have range of values [123]. The discrete variables are those which can take only certain type of variables like total number of persons that can fit into a bus.

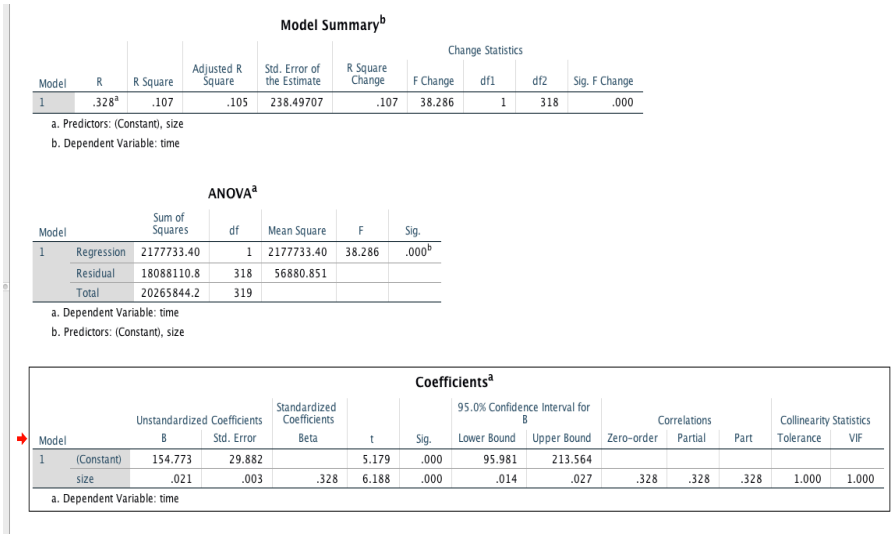
Moving on, as the study has the variables are continuous so regression is perfectly apt, However, two important challenges observed that is firstly, there exists multicollinearity which is deduced from coefficients table and all the metrics have significance greater than 0.05 which is contradicting the models summary results. So, to mitigate this challenge the deeper insight into multicollinearity should be look upon. Some solutions to deal with multicollinearity are:

Case 1: **increase the sample population** This cannot be possible as the population that is considered is fixed and cannot be increased as we could not conduct another experiment. We have taken 32 participants and we are confident that they are sufficient.

Case 2: **Type1 step wise regression** Type 1 step wise is to understand the effects of regression. The type 1 step wise regression has a specialty in only recognizing the independent variable which shows significant variation with the dependent variable and does not include which do not show variance in dependent variable.

Important identification: From the data shown in the below figure 5.3 when the step wise multiple regression is performed only the size metric is significantly contributing to the model. The significance p = 1.8796E-9 which is p<0.05 and it has the R-squared value from the model which is 0.107 that means 10.7 of the variability in the independent variable is explained by the size metric. The independent variable that are not showing significant contribution are excluded. Other proofs to consider this step wise method of regression analysis model are as follows:

- The overall variation in dependent variable from Equation1 is 12.6% and here 10.7% of the 12.6% variance is shown by the size itself.
- The unstandardized beta value of size for 1 unit increase in the number of bytes of size the time increase by 0.021 seconds.
- The standardized beta value here for size metric is 0.328, the value is positive and indicating the size metric is contributing to the dependent variable time.
- the tolerance should be greater than 0.1 and VIF should be less than 10 which is true in this model.



Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics				
					R Square Change	F Change	df1	df2	Sig. F Change
1	.328[a]	.107	.105	238.49707	.107	38.286	1	318	.000

a. Predictors: (Constant), size
b. Dependent Variable: time

ANOVA[a]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	2177733.40	1	2177733.40	38.286	.000[b]
	Residual	18088110.8	318	56880.851		
	Total	20265844.2	319			

a. Dependent Variable: time
b. Predictors: (Constant), size

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B		Correlations			Collinearity Statistics	
		B	Std. Error	Beta			Lower Bound	Upper Bound	Zero-order	Partial	Part	Tolerance	VIF
1	(Constant)	154.773	29.882		5.179	.000	95.981	213.564					
	size	.021	.003	.328	6.188	.000	.014	.027	.328	.328	.328	1.000	1.000

a. Dependent Variable: time

Figure 6.3: SPSS statistical tool helps to statistically calculate many different statistic's based on the convenience of the researcher.

Case 3: **Type 2 step wise regression** The Type 2 step wise regression is facing the actual multicollinearity itself which is basically by removing the variables which cause the multicollinearity in the descending order one by one [124] [125] [126].
We removed each variable one by one and observed the change in the multicollinearity among the independent variables. The independent variables are removed in the descending order of VIF value. The order of removal is Lines Of Code then Number of tags then size then compress size then anchor tag then <p> then lastly div tag.

Case 1: When Lines of code metric is removed from the independent variable list:

- Compress size with p=0.044 p<0.05 show significance. Unlike in the first regression where no independent variable p value is less than 0.05. The R-square value of this model is just slightly differed from 12.6 % to 12.4 %.

Case 2: When Loc and number of tags both are removed from the independent variable list

- The compress size showed higher significance that is from p=0.044 in previous case to p=0.039 which means lesser the p value higher is the significance.

Case 3: When only tag metric is removed and all other metric variables are included:

- No independent variable that is metrics have significance p<0.05 so, don't this removal does not show impact.

Case4: When only Size metric is removed and all other metric variables are included

- No significance is identified and all the metrics have p value>0.05 so no use by removing this size metric as it does not show impact.

Case 5: When size, tags and loc metric independent variables are removed and excluded from the list:

- Once again here the compress size is less than 0.05 p=0.029 which is the least value till now which indicates more significance.
- Along with compress size, <Div> tag also has shown significance of p=0.027 much less than the compress size value.
- The model summary is very slightly differing from 12.6 % to 12.2 %.

Case6: When four metrics size compress size number of tags and loc are excluded from the independent variable list

- The div tag show p value= 0.021 and it is much lesser than the previous model that is p=0.027 and it rejects the null hypothesis as well.

Case 7: When only size and compress size are excluded and all other metrics are included:

- None of the metrics among the included independent variable list show significance and all of them have p>0.05

Case8: When size, Compress size, Tags, lines of code and anchor <a> tag is excluded from the independent variable list

- The div tag show p= 0.000037 which means null hypothesis can be rejected and 1 unit increase in the div tag increase the time by 4.215 seconds. This relation is deduced from the unstandardized B value of div tag.
- The tolerance is <0.1 an VIF is greater than 1 for all three metrics included.

Case9: When only depth and div tags are included in the independent variable list

- The div tag s how much lower p value and higher significance that is p= 0.000002 p<0.05 and null hypothesis can be rejected.
- When only these two metrics are included the tolerance is less than 0.1 and VIF is greater than 1 which is true and valid for the model to be successful.

6.3 Accuracy vs Metric independent variables

This Accuracy vs metric analysis will help to identify any metrics are influencing the accuracy of the results that is are there any metrics that impact in answering the questions correctly. The linear discriminant analysis is performed here because the dependent variable in this case is categorical that means it is fixed the answer is either true or the answer is false there is not third option in this scenario [127] [128] [129]. So when the dependent variable is categorical and independent variable is continuous one way to understand the patterns among the independent and dependent variables is by applying linear discriminant analysis LDA [130]. This LDA can be performed using the statistical tool SPSS. If the answer given by the participant is correct then it is indicated numerically by 1.00 and if it is a wrong answer, then the value is 0.00. As we have only one measure entity accuracy hear we can apply the linear discriminant analysis. The linear discriminant analysis is a method that is used and applied in the statistics to understand and recognize if there are any patterns that impact the outcomes. We performed the linear regression analysis using the accuracy and the metrics to understand if the metrics show any impact while answering. When the analysis is performed we had two case one to include all the independent variables together at once and the other is to include the independent variables step by step.

Case 1: Entering all independent variables together

When all the independent variables are included at once the results are not promising as the model is not significant. Among the drawn conclusions from the analyzed data one important noticeable information is the significance of the prediction model which is shown in the below table 6.5 the p value is >0.05 and the significance test fails.

Wilks' Lambda				
Test of Function(s)	***Wilks' Lambda***	***Chi-square***	***df***	***Sig.***
1	.961	11.349	8	.183

Table 6.5: The Wilks' Lambda function helps to notice significance of the model using the Linear Discriminant analysis.

If observed from the final column of table 6.6 significance sig only the tag and <p> tag show the significance value p<0.05. The tests of equality of group means primarily address mean score. If the mean score significantly differs among the participants who answer the results correctly and the participants who answer incorrectly this information can be deduced from the table. However, in this case only tag and <p> tag show that the ability to answer correctly is significantly differ from the ability to answer incorrectly their values are <0.05.

Tests of Equality of Group Means					
	Wilks' Lambda	***F***	***df1***	***df2***	***Sig.***
depth	.999	.375	1	292	.541
size	.991	2.668	1	292	.103
compress	.996	1.208	1	292	.273
tags	.998	.723	1	292	.396
loc	.996	1.140	1	292	.286
div	.979	6.132	1	292	.014
anchor	1.000	.001	1	292	.973
p	.986	4.163	1	292	.042

Table 6.6: The test of equality of group means displaying that all the significance values of individual independent metrics.

This model is not good prediction model so we avoided using this as a base for our studies final conclusion which is to understand any of the metrics does show influence in answering the test inputs correctly. So, then better alternative way is to perform step wise analysis.

Case2: Entering the independent variables step wise

Using the step wise the results are very promising wand we could notice that the prediction model significance test is a pass as the sig value in the below table 6.7 show $p<0.05$ which means there are metrics that does influence the correctness and answering accurately.

Wilks' Lambda				
Test of Function(s)	***Wilks' Lambda***	***Chi-square***	***df***	***Sig.***
1	.965	10.470	2	.005

Table 6.7: The Wilks' Lambda function helps to notice significance of the model using the Linear Discriminant analysis.

The metrics div and lines of code are the only metrics independent variables that predict the significance of the model and both of them from table 6.8 show the significance value less than 0.05 for div p= 0.014 and lines of code p=0.005. These two metrics div tag and number of lines of code influence the outcome to answer the questions correctly.

Variables Entered									
Step	***Entered***	***Wilks' Lambda***							
		Statistic	***df1***	***df2***	***df3***	***Exact F***			
						Statistic	***df1***	***df2***	***Sig.***
1	div	.979	1	1	292.000	6.132	1	292.000	.014
2	loc	.965	2	1	292.000	5.331	2	291.000	.005

Table 6.8: The test of equality of group means displaying that all the significance values of individual independent metrics.

6.4 Use of experiment/ Research Contribution

RQ.3b Among the selected metrics that are applied on the test data during the experiment, which of these predictors is/are best?
By performing above analysis by applying various techniques like multiple regression for overall model, then stepwise regression [131] [132] [133] [134] to avoid the multicollinearity [125] and also removing each independent variable to see which metric is showing significance by doing so some very important conclusions are draw these are addressed below
Important conclusions:

1. When initially all the independent variables are included and multiple regression is performed
   (a) size is highly correlating with time.
   (b) In the coefficients table in which none of the independent variables show $p<0.05$.
   (c) There is a High multicollinearity among the metrics independent variables, VIF values are very high and which indicates there is high correlations among the metrics. So, identifying one single metric that influence test data is a hard task.
2. The step wise regression model summary shows 10.7% out of 12.6 % of the variance in dependent variable is explained by the size metric itself.
3. When the independent variables are removed one by one in the descending order of VIF.
   (a) The compress size is showing $p<0.05$.
   (b) Independent variables compress size ($p=0.029$) and <Div> tag ($p=0.027$) show variance in time.
4. From the step wise linear discriminant analysis, the div tag and number of lines of code impact the participants accuracy in answering the results accurately.

6.5 Summary of findings from Experiment

From the Coefficients table 6.4 yes, there is multicollinearity it wasn't a big surprise, we reduced the multicollinearity by removing the highly correlating metrics one by one. We found that size, Compress size and Div tag show positive significance and this result is matching with the correlations table 6.1. So, the results obtained by reducing the multicollinearity is extend-able to other studies as size, compress size and div tag show variance in time. From accuracy vs metrics when we applied linear discriminant analysis we found that div tag and number of lines of code impact the participants accuracy in selecting correct output.

Chapter 7

Discussion and Limitations

7.1 Discussion

This chapter 7 gives a comprehensive discussion on what has been found in the study. The results from the literature review act as a starting point for considering the test input and observe if the test input can be suitable for the study. So, the literature review results play a cohesive role to move on to the experiment chapter. The experiment is conducted in a controlled environment with the BTH students as subjects. This chapter includes the discussion made on both the results from literature and experiment which shall answer the research question formulated for the research.

Initially the literature review is performed to understand what is human oracle costs, the literature specifically in this area is considerably very low. Thus as the research gap is identified it is important to consider the human oracle costs and the strategies to reduces these costs. To reduce these human oracle costs the factors/ metrics that impact the test data are investigated. To find such metrics that impact the test data a snowballing is performed to understand if there is any background in this specific metrics in test data area.

7.1.1 Answering the Research Questions

RQ.1 To find metrics that are good predictors of human oracle costs the literature is reviewed to understand the metrics on test data?
Discussion: The metrics always depend on the type of programming language used as test input, input in our study used is HTML. The literature on metrics associated with test data are considerably very low and those metrics that are discussed about the test data metrics relating to object oriented paradigms, and as not all these metrics can be applied to the other procedural language's and web development languages so, among these software metrics that are applied on test data, some metrics are chosen that are very general and can be applied to all the programs irrespective the language barrier. The size is a general measures and can be applied on any test input without any language barrier so size metric is considered. Size is supported as a general metric in both metrics applied on test data literature and also literature review on code metrics.

RQ.2: Are there any existing metrics used in the literature, that can potentially measure the human comprehensibility?
Discussion: Yes, there are metrics that influence the comprehensibility of the human.

Writing a code which is readable and understandable to existing developers and new developers who would like to reuse the code is very important. If the original code can be replaced with shorter code version then the number of lines and depth of the code vary/changes. The source code is also in the form of text and sometimes there is a lot of repetition of text, feature based similarities white spaces and similar kind of code repeating multiple number of times, this influences the overall size of the document. It discusses using compression to calculate a similarity distance metric, motivated by the fact that the compression size is an approximation of Kolmogorov complexity. So, the compress size is different from the size and it is always lesser in bytes. The compress size can be applied to any programming language irrespective of type. TTR Table tag ratio which is the estimation of total number of table tags to the tags in the HTML document to classify the web pages. HTML tags in the Hypertext documents is quite rich and modular, he supports that much more information can be learned by analyzing the use of HTML tags. From the literature we found that three important metrics influence the comprehensibility of test data/ source code they are Tags in HTML, depth of the source code and the compression size of text.

RQ.3a To identify if the metrics inspired by source code (code metrics) are usable as good predictors in estimating human oracle costs?
Discussion: we performed an extensive search beyond the test data metrics as the literature is considerably low, so we looked for some possible code metrics that can be applied on test data. The keywords like code metrics, software metrics, comprehensibility, understandability, HTML test inputs, HTML test sets and test data generation techniques. From these keywords the literature is gathered and we found some metrics like number of tags in HTML and depth of the nodes in the HTML tree these two metrics are noticed in the literature and for this study we considered they might have some potential impact so these two metrics are taken into account.

7.1.2 Experiment test results showing which metric is a good predictor of Human oracle costs

RQ.3b To identify any of the metrics which are applied on the test data which are cost effective and are good predictors of the human oracle costs?
Discussion: *Time vs metrics:* Size among the independent variable show significant amount of variation with dependent variable time. When Multicollinearity is taken into account and reduced, new metrics show significance like Compress size and tag show the significant variation in the time dependent. The tag helps to define the sections in HTML. The <Div> tag is impacting the time and the reason behind this significance is to look for each and every section and within the tag each, section might be different so it is time consuming to go through the entire test inputs. As the depth of the nodes that is level increase the complexity of the test input increases so it will impact time taken to answer the test input.

Accuracy vs metrics: After performing the Step wise linear discriminant analysis LDA the metrics <Div> tag and number of lines of code show impact to answer the test inputs correctly. Which is not surprising because the participants from the interview mentioned the lines of code is very important factor while working on the test inputs. The <Div>

tag constitutes all the important classes and it is a very important as the tag helps to divide the HTML into different sections and include different classes and Id's for each section,it differs from one section to another in some cases so, to go through all the tags is important for the tester to check the correctness of the output.

7.2 Limitations and Threats to validity

7.2.1 Limitations

Although the thesis study is carefully presented we are aware of some unavoidable limitations and shortcomings these are addressed below:

a. We could apply mean centering instead of type 2 step-wise regression, this is a limitation as we do not know what results would have turn out when the mean cantering technique is applied.

b. 1 participant in interview 2 suggested, the position of the mutant applied on the test input influenced the concentration level and brings negligence into picture however, we did not consider this criterion this can be limitation.

c. The color of the font is directly matching in some case with the background which is the feedback mentioned by 2 participants, which is a limitation that should have been avoided while this study is performed.

d. Different versions of HTML when combined then that might influence the study like HTML1, HTML5 mixed in single test input.

e. Participants involved in the study are not industrially experienced and this influence the feedback and answers given by them. However, unavailability of industrial contacts allows us to stick to only to this format.

f. The population sample is relatively medium neither large nor small which might impact on the type of analysis being performed for example the multicollinearity would have varied when the participants sample size is much bigger.

g. The experiments are conducted in the laboratory in the university in which participants are subjected to do some tasks this might impact their behavior however we tried to control the experiment to reduce such disorientation's.

h. In practice it is impossible to have control over all the existing variables. Even though the strength lies in controlling the variables in the experiment research studies this is not practically possible to reach such targets.

i. A HTML test would have been conducted before the participants participated in the experiment this is a limitation for this study.

7.2.2 Threats to validity

It is very important to notice and address the validity threats for the research design taken and the results obtained from it, this helps to address the quality of the research study. For any study the critical task is to analyze and mitigate the threats to validity [135] [136]. The chosen research method- experiment, experiment is the primary research method chosen for the study. So, by employing the research method and generating the result is not just enough, the task to identify the challenges and threats and mitigates these challenges is very important to determine the quality of the study. The experiments which generate a quantitative data type have results prone to more validity threats. For empirical studies such as this one, there exists four major validity threats Conclusion validity, construct validity, internal validity and external validity [137]. This section primarily stress on the validity threats that are relevant to this study, along with mitigation strategies that are implemented to the best of our knowledge, the mitigation strategies applied are based on [136].

Construct Validity

The rate at which the measures that are made accurately represent what is need to be investigated, what percent rate is the cause and effect relation true [137]. In the conducted research there are two possibilities of construct validity threats. Misinterpretation of questions during group discussion might lead to collection of irrelevant data. This threat is attempted to be mitigated by taking proper care in formulating the interview questions and conducting the interview by following the proper guidelines.

Another Threat is Some of the interview questions are Leading questions i.e., how they are phrased leads interviewees to answer in different ways. To reduce this threat a voting scheme is established for the answers different answers that are generated for the same question, the leading questions are some times very desirable so we have given always importance to these questions and use them only when there is a deliberate purpose to extract more information/opinions about the same question.

We made sure that the interview questions are formulated in alignment with the research objectives. Further, took guidance of our supervisor for getting the feedback on the questions and reformulating them. Identifying biased answers during the group interviews is another construct validity threat. This can be occurred due to misinterpretation of questions or phrases. The revision of such answers is done by eliminating such answers that are not related to the research questions. However, we cannot mitigate this threat completely in a fear loosing some important information. However, this can compensated by trying to achieve as much as data from feedback form after the interview.

Although in this research study the use of general and specific terms in the metrics would differ in different contexts. To mitigate this threat, when the metrics are addressed with these terms a clear insight is given about these terms to reduce the misconception from actually what the term means in this study.

Another Threat is are the metrics you measured (time taken and accuracy) really an indication of comprehensibility? To answer this yes, the keywords used to extract the literature is certainly relevant to comprehension of test data, source code and text. The gathered literature support that these metrics are some among many which are suitable

for measuring the comprehension of the source code/text, Size and Compress size are very general and can be applied to any program, Depth of node is similar to depth of HTML (since HTML is tree like structure), number of tags is also specific to HTML.

Internal Validity

If there is a statistical significance among the independent variables and dependent variable, how sure are these results answering that treatments actually impacted the outcomes [129]. In this study to mitigate this threat we addressed it in two ways. Firstly, our initial Summary model does show statistical significance with $p<0.05$ however, yet we did not draw our conclusions only this analysis from the results, and yes the results were contradicting due to the multicollinearity existence so we applied different techniques like step-wise multiple regression and removing highly VIF indicating variables to see which metric is a good predictor and the results were promising and metrics were showing significant variance.

Secondly, while estimating the accuracy vs metric first when entire independent variable set is considered for the model,model is insignificant and we further investigated deeper by applying the step wise then the results are promising as the metrics like tag and lines of code does show significance so, from this study the first results does not mean that they are final and there is no significant contribution further investigation is always necessity to attain quality.

In our study we came across some internal validity threats like inappropriateness in choosing the literature, misconception of data, improper selection of participants for the experiment. In order to avoid the threats mentioned the following steps have been taken for a valid output. The selection of participants for the experiment affects the data that is collected. Here all the participants we selected are students having expert/intermediate level knowledge in HTML. The unsubstantial data analysis miss tracks our result in wrong path. To overcome this risk, it is quickly discussed with the supervisor to approve the validity of the findings.

External Validity

These threats are related to generalisability i.e., to identify whether the results can be generalized to larger population outside the research scope [137]. In a controlled experiment the results occur depending on the treatment, objects and environmental settings used. one such threat is not having proper environmental settings. This threat is mitigated as we booked our college computer laboratories which have closed setup of all the required objects like computers, proper Internet connection etc.

Another important external validity threat is timing, During the pilot study we identified that participants not being able to complete the test within the prescribed set time. After both the pilot studies for the main experiment we have increased the time limit. However, this threat cannot be mitigated completely as it also differs from participant to participant depending on individual expertise and capability.

Another such validity threat is being able to make the participants attend the experiment so that the laboratory booking and participants availability are in sink. To avoid such threats where absence of participants can happen which occurred during our pilot

study. So, for the main experiment, we booked four different for two days one in morning and one in afternoon and asked the participants to sign-up as per their availability.

The ability to generalize the results and forecast them outside the current study boundary [137]. Since the findings are relevant and show impact on the entire HTML test input type which is the only test input type used and mutation were applied on, such external validity threats might encounter to generalize the results to out of scope studies. This threat is mitigated by explaining clearly why the HTML test input type is taken and why other languages are avoided. Metrics like size is similar in other programming languages and the results from this study also suggest size is a major contributor in showing the significant variances.

The regression model show low R square value which means we cannot over claim from R Square. The low R square values clearly indicates there are some variables clearly missing which we haven't taken into account these values can be metrics of any type. so, concluding/over-claiming the results from r-square would be a threat to this study.

There is multicollinearity in the coefficients table. when we selected the examples the constraint was to avoid collinearity as fas as possible and yes we selected test input with metrics showing significant variation but we were unable to avoid multi collinearity. For us multicollinearity is not a big surprise, it was always there from the beginning and the ideas is to pick the test data that minimizes the multi collinearity. To minimize the multicollinearity we have removed more collated independent variables step by step from the model. We found that size, Compress size and Div tag show positive significance and this result is matching with the correlations table 6.1. So, the results obtained by reducing the multicollinearity is extend-able to other studies as size, compress size and tag show variance in time. Similarly the tag and lines of code are impacting the participants accuracy which can be extended to further studies to find new metrics that impact comprehension.

Conclusion validity

Is the treatment chosen for the study is correct one and how related is the treatment to the outcome [137]. This threat can be noticeable in this study as there are more than one independent variables present in the study that are manipulated by us. To mitigate this study, we made sure that the metrics show significance variation in the test inputs and the information about their variation among the test inputs which are selected for the experiments are very clearly stated in the document when and where is needed. As the HTML is the only test input type used in this study all the metrics which are relevant and can be applied to the HTML are searched and selected very carefully.

Repeat-ability

Is the study repeatable and in the sense trustworthy to look through while implementing similar further on, is the study reliable? This is a concern in every research study. In this research the experiments conducted is at university level and not primarily in the industry so it can be repeated with different technologies using different programming languages to see what metrics that that particular programming languages might influence and show statistical significance. Moreover, the decision's taken and actions performed throughout the research is being monitored and mentored by the supervisor with

his expertise and careful suggestions. The study has improved in delivering better quality results which make this study both reliable, which can be observed from the results as there are metrics that show significance and also repeatable as we primarily focused and limited to specific HTML test input it has a large scope to expand for future research work.

Scope

This is a unique study under taken by us to perform and deliver a better results, thus based on complexity in the problem domain there is a risk for misinterpretation. To reduce this risks, we stated and primarily stressed our only goal is to find the metrics that are suitable and good predictors of comprehensibility of test data. There exists some metrics that show variation in time and accuracy, to draw these conclusions we followed a very systematic procedure. The primary goal is stated very clearly what we are going to measure and when and where ever needed. We also mentioned how we are going to achieve this target by applying experiment protocols and step by step implementation so there is lower chance of misconception about the project.

Chapter 8

Conclusions and Future Work

8.1 Conclusions

With the advent in software testing over the years the study about test data generation more specifically complete automated test data generation is becoming more challenging specifically the cost associated with identifying the correctness of the output for the given test inputs. Our study primarily focused to identify some or any of the metrics when applied to the test inputs can help predict human oralce costs and these identified metrics does show significant impact on the test input selected. To do this study the metrics related to the test data are identified from the literature review. This chapter 7 addresses the conclusion drawn from the study and future work.

In this study along with the literature review to identify the relevant metrics to be applied on the test input an experiment is also conducted to understand which metric or metrics impact the test data. 2 pilot studies are run before the real experiment is conducted to understand if the test inputs taken are apt for the real experiment. The feedback from the pilot studies are highly helpful to improvise the final experiment. A pre-questionnaire is conducted to know if the participant has the experience in the test inputs and a post-questionnaire is conducted to gain their feedback about the experiment and the challenges they faced are gathered. After the experiment is finished a group interview is conducted to view on the participant's perspective on any new metric which they believe from the experiment which shows significant impact on the test input.

After the entire experiment protocol is implemented the entire experiment data is gathered the regression analysis is performed to understand among the all the selected metrics which metrics show significant variation, which one/ are the good predictors of human oracle cost are concluded from the regression analysis. However, the data obtained in the regression analysis does show significance of $p<0.05$ that is null hypothesis is rejected but in the coefficient table, observed carefully all the 8 metrics independent variables show the significant variance to the dependent variable $P>0.05$ which contradicts from the results obtained from the Model summary table. All the metrics have positive correlation with the test data. Even though the model is significant due to multi collinearity among variables with each other.

To reduce the multi collinearity the step wise regression is performed and from Type 1 step wise regression Size with 10.7 % of R-square value shows significant variance in the time dependent variable. In type 2 step wise regression by removing each metric independent variable one at a time then Compress size and tag significance variation in the time dependent variable.

From the accuracy vs metric combinations even though when all the metrics are combined and linear discriminant analysis is performed the results are not promising as the p value =0.183 which means no metric is influencing the correctness of the output and yet we did not stop our study here we tried different techniques to see how the metrics respond to the accuracy or correctness this additional work has always been promising. however, when step Wise linear discriminant analysis is implemented the results show <Div> tag and the number of lines of code from the prediction model show significance influence in answering the questions correctly.

8.2 Future Work

- Future work can be implemented by considering different programming language techniques and see how the metrics influence the comprehensibility of the test data.
- By considering HTML, CSS and JavaScript all together and with experienced people in industries if the research is performed then more metrics can be explored as different web technologies are used and new metrics can be identified those that impact the comprehensibility of the test data.
- The oracle problem should be focused more in terms of Web technologies perspective as there is very little literature to support and understand the metrics that can be applied to test data specifically if the test inputs that are used in the experiment belong to diverse range of core web technologies like HTML, CSS, Java servlets and so on.
- Strengthening the research in the area of oracle problem is very important as the oracle problem is addressed in the literature and among them very few primarily relate the oracle problem with the test data generation so there is a need for future work as this defines the way we look at the test data generation itself.
- We measured Time vs metric significance and accuracy vs metric significance. However, both time and accuracy can be combined and analyzed with metrics in future.
- A similar study can be implemented by considering size as the only independent variable in different programming paradigms to see how they interact.
- same study can be replicated in industry by preforming various desk experiments further would enhance the study to gather more reliable information.

References

[1] L. Manolache and D. G. Kourie, "Software testing using model programs," *Software: Practice and Experience*, vol. 31, no. 13, pp. 1211–1236, 2001.

[2] E. T. Barr, M. Harman, P. McMinn, M. Shahbaz, and S. Yoo, "The oracle problem in software testing: A survey," *IEEE transactions on software engineering*, vol. 41, no. 5, pp. 507–525, 2015.

[3] S. R. Dalal and A. A. McIntosh, "When to stop testing for large software systems with changing code," *IEEE Transactions on Software Engineering*, vol. 20, no. 4, pp. 318–323, 1994.

[4] R. Feldt and S. Poulding, "Finding test data with specific properties via metaheuristic search," in *2013 IEEE 24th International Symposium on Software Reliability Engineering (ISSRE)*, pp. 350–359, IEEE, 2013.

[5] A. Memon, I. Banerjee, and A. Nagarajan, "What test oracle should i use for effective gui testing?," in *Automated Software Engineering, 2003. Proceedings. 18th IEEE International Conference on*, pp. 164–173, IEEE, 2003.

[6] C. D. Nguyen, A. Marchetto, and P. Tonella, "Automated oracles: An empirical study on cost and effectiveness," in *Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering*, pp. 136–146, ACM, 2013.

[7] A. M. Memon, M. E. Pollack, and M. L. Soffa, "Automated test oracles for guis," in *ACM SIGSOFT Software Engineering Notes*, vol. 25, pp. 30–39, ACM, 2000.

[8] A. Shahbazi, *Diversity-Based Automated Test Case Generation*. PhD thesis, University of Alberta, 2015.

[9] S. Mirshokraie, A. Mesbah, and K. Pattabiraman, "Jseft: Automated javascript unit test generation," in *2015 IEEE 8th International Conference on Software Testing, Verification and Validation (ICST)*, pp. 1–10, IEEE, 2015.

[10] P. McMinn, "Search-based software test data generation: A survey," *Software Testing Verification and Reliability*, vol. 14, no. 2, pp. 105–156, 2004.

[11] M. Harman, Y. Jia, and Y. Zhang, "Achievements, open problems and challenges for search based software testing," in *2015 IEEE 8th International Conference on Software Testing, Verification and Validation (ICST)*, pp. 1–12, IEEE, 2015.

[12] C. Mao, "Harmony search-based test data generation for branch coverage in software structural testing," *Neural Computing and Applications*, vol. 25, no. 1, pp. 199–216, 2014.

[13] K. Gao, T. M. Khoshgoftaar, and A. Napolitano, "Impact of data sampling on stability of feature selection for software measurement data," in *2011 IEEE 23rd International Conference on Tools with Artificial Intelligence*, pp. 1004–1011, IEEE, 2011.

[14] A. Memon and Q. Xie, "Using transient/persistent errors to develop automated test oracles for event-driven software," in *Proceedings of the 19th IEEE international conference on Automated software engineering*, pp. 186–195, IEEE Computer Society, 2004.

[15] M. D. Davis and E. J. Weyuker, "Pseudo-oracles for non-testable programs," in *Proceedings of the ACM'81 Conference*, pp. 254–257, ACM, 1981.

[16] S. Afshan, P. McMinn, and M. Stevenson, "Evolving readable string test inputs using a natural language model to reduce human oracle cost," in *2013 IEEE Sixth International Conference on Software Testing, Verification and Validation*, pp. 352–361, IEEE, 2013.

[17] S. Anand, E. K. Burke, T. Y. Chen, J. Clark, M. B. Cohen, W. Grieskamp, M. Harman, M. J. Harrold, P. McMinn, *et al.*, "An orchestrated survey of methodologies for automated software test case generation," *Journal of Systems and Software*, vol. 86, no. 8, pp. 1978–2001, 2013.

[18] P. Ciancarini, A. Rizzi, and F. Vitali, "An extensible rendering engine for xml and html," *Computer Networks and ISDN Systems*, vol. 30, no. 1, pp. 225–237, 1998.

[19] X. Guo, M. Zhou, X. Song, M. Gu, and J. Sun, "First, debug the test oracle," *IEEE Transactions on Software Engineering*, vol. 41, no. 10, pp. 986–1000, 2015.

[20] S. Liu, *Generating test cases from software documentation.* PhD thesis, McMaster University, 2001.

[21] T. Kanstrén, "Program comprehension for user-assisted test oracle generation," in *Software Engineering Advances, 2009. ICSEA'09. Fourth International Conference on*, pp. 118–127, IEEE, 2009.

[22] B. Canou and A. Darrasse, "Fast and sound random generation for automated testing and benchmarking in objective caml," in *Proceedings of the 2009 ACM SIGPLAN workshop on ML*, pp. 61–70, ACM, 2009.

[23] Q. Yang, J. J. Li, and D. M. Weiss, "A survey of coverage-based testing tools," *The Computer Journal*, vol. 52, no. 5, pp. 589–597, 2009.

[24] G. Fraser and A. Arcuri, "Evolutionary generation of whole test suites," in *2011 11th International Conference on Quality Software*, pp. 31–40, IEEE, 2011.

[25] F. Pastore, L. Mariani, and G. Fraser, "Crowdoracles: Can the crowd solve the oracle problem?," in *2013 IEEE Sixth International Conference on Software Testing, Verification and Validation*, pp. 342–351, IEEE, 2013.

[26] M. Harman, S. G. Kim, K. Lakhotia, P. McMinn, and S. Yoo, "Optimizing for the number of tests generated in search based test data generation with an application to the oracle cost problem," in *Software Testing, Verification, and Validation*

Workshops (ICSTW), 2010 Third International Conference on, pp. 182–191, IEEE, 2010.

[27] S. Poulding and R. Feldt, "Generating structured test data with specific properties using nested monte-carlo search," in *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*, pp. 1279–1286, ACM, 2014.

[28] M. Harman, P. McMinn, M. Shahbaz, and S. Yoo, "A comprehensive survey of trends in oracles for software testing," *University of Sheffield, Department of Computer Science, Tech. Rep. CS-13-01*, 2013.

[29] S. Poulding and R. Feldt, "The automated generation of human-comprehensible xml test sets," in *Proc. 1st North American Search Based Software Engineering Symposium (NasBASE)*, 2015.

[30] S. Afshan, "Search-based generation of human readable test data and its impact on human oracle costs," 2013.

[31] C. Hart, *Doing a literature review: Releasing the social science research imagination*. Sage, 1998.

[32] T. J. Ellis, "The literature review: The foundation for research," 2006.

[33] C. Wohlin, "Guidelines for snowballing in systematic literature studies and a replication in software engineering," in *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering*, p. 38, ACM, 2014.

[34] S. L. Pfleeger, "Experimental design and analysis in software engineering," *Annals of Software Engineering*, vol. 1, no. 1, pp. 219–253, 1995.

[35] C. W. Knisely and K. I. Knisely, *Engineering Communication*. Cengage Learning, 2014.

[36] D. Coleman, D. Ash, B. Lowther, and P. Oman, "Using metrics to evaluate software system maintainability," *Computer*, vol. 27, no. 8, pp. 44–49, 1994.

[37] A. Meneely, B. Smith, and L. Williams, "Validating software metrics: A spectrum of philosophies," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 21, no. 4, p. 24, 2012.

[38] R. Harrison, L. Samaraweera, M. R. Dobie, and P. H. Lewis, "Estimating the quality of functional programs: an empirical investigation," *Information and Software Technology*, vol. 37, no. 12, pp. 701–707, 1995.

[39] N. E. Fenton and M. Neil, "Software metrics: successes, failures and new directions," *Journal of Systems and Software*, vol. 47, no. 2, pp. 149–157, 1999.

[40] L. Rosenberg, T. Hammer, and J. Shaw, "Software metrics and reliability," in *9th International Symposium on Software Reliability Engineering*, Citeseer, 1998.

[41] A. B. De Carvalho, A. Pozo, and S. R. Vergilio, "A symbolic fault-prediction model based on multiobjective particle swarm optimization," *Journal of Systems and Software*, vol. 83, no. 5, pp. 868–882, 2010.

[42] P. Viqarunnisa, H. Laksmiwati, and F. N. Azizah, "Generic data model pattern for data warehouse," in *Electrical Engineering and Informatics (ICEEI), 2011 International Conference on*, pp. 1–8, IEEE, 2011.

[43] J. W. Palmer, "Web site usability, design, and performance metrics," *Information systems research*, vol. 13, no. 2, pp. 151–167, 2002.

[44] P. Leite, J. Gonçalves, P. Teixeira, and Á. Rocha, "Assessment of data quality in web sites: towards a model," in *Contemporary Computing and Informatics (IC3I), 2014 International Conference on*, pp. 367–373, IEEE, 2014.

[45] T. Repasi, "Software testing-state of the art and current research challanges," in *Applied Computational Intelligence and Informatics, 2009. SACI'09. 5th International Symposium on*, pp. 47–50, IEEE, 2009.

[46] M. K. Debbarma, N. Kar, and A. Saha, "Static and dynamic software metrics complexity analysis in regression testing," in *Computer Communication and Informatics (ICCCI), 2012 International Conference on*, pp. 1–6, IEEE, 2012.

[47] U. Raja, D. P. Hale, and J. E. Hale, "Modeling software evolution defects: a time series approach," *Journal of Software Maintenance and Evolution: Research and Practice*, vol. 21, no. 1, pp. 49–71, 2009.

[48] P. Luchscheider and S. Siegl, "Test profiling for usage models by deriving metrics from component-dependency-models," in *2013 8th IEEE International Symposium on Industrial Embedded Systems (SIES)*, pp. 196–204, IEEE, 2013.

[49] O. Signore, "A comprehensive model for web sites quality," in *Seventh IEEE International Symposium on Web Site Evolution*, pp. 30–36, IEEE, 2005.

[50] V. R. Basili, R. W. Selby, and T. Phillips, "Metric analysis and data validation across fortran projects," *IEEE Transactions on Software Engineering*, no. 6, pp. 652–663, 1983.

[51] V. R. Basili, L. C. Briand, and W. L. Melo, "A validation of object-oriented design metrics as quality indicators," *IEEE Transactions on software engineering*, vol. 22, no. 10, pp. 751–761, 1996.

[52] G. Manduchi and C. Taliercio, "Measuring software evolution at a nuclear fusion experiment site: a test case for the applicability of oo and reuse metrics in software characterization," *Information and Software Technology*, vol. 44, no. 10, pp. 593–600, 2002.

[53] O. P. Dias, I. C. Teixeira, and J. P. Teixeira, "Metrics and criteria for quality assessment of testable hw/sw systems architectures," *Journal of Electronic Testing*, vol. 14, no. 1-2, pp. 149–158, 1999.

[54] R. Harrison, L. Samaraweera, M. R. Dobie, and P. H. Lewis, "An evaluation of code metrics for object-oriented programs," *Information and Software Technology*, vol. 38, no. 7, pp. 443–450, 1996.

[55] P. Devanbu, S. Karstu, W. Melo, and W. Thomas, "Analytical and empirical evaluation of software reuse metrics," in *Proceedings of the 18th international conference on Software engineering*, pp. 189–199, IEEE Computer Society, 1996.

[56] T. Hall and N. Fenton, "Implementing effective software metrics programs," *IEEE software*, vol. 14, no. 2, p. 55, 1997.

[57] N. Ramasubbu and R. K. Balan, "Overcoming the challenges in cost estimation for distributed software projects," in *Proceedings of the 34th International Conference on Software Engineering*, pp. 91–101, IEEE Press, 2012.

[58] S. A. Mengel and J. V. Ulans, "A case study of the analysis of novice student programs," in *Software Engineering Education and Training, 1999. Proceedings. 12th Conference on*, pp. 40–49, IEEE, 1999.

[59] K. Gao, T. M. Khoshgoftaar, and A. Napolitano, "Impact of data sampling on stability of feature selection for software measurement data," in *2011 IEEE 23rd International Conference on Tools with Artificial Intelligence*, pp. 1004–1011, IEEE, 2011.

[60] M. Jiang, M. A. Munawar, T. Reidemeister, and P. A. Ward, "System monitoring with metric-correlation models," *IEEE Transactions on Network and Service Management*, vol. 8, no. 4, pp. 348–360, 2011.

[61] D. Walker and A. Orooji, "Metrics for web programming frameworks," in *Proceedings of the International Conference on Semantic Web and Web Services, Las Vegas, NV*, 2011.

[62] H. Berghel, "Using the www test pattern to check html client compliance," *Computer*, vol. 28, no. 9, pp. 63–65, 1995.

[63] M.-H. Lee, Y.-S. Kim, and K.-H. Lee, "Logical structure analysis: From html to xml," *Computer Standards & Interfaces*, vol. 29, no. 1, pp. 109–124, 2007.

[64] A. Andrić, V. Devedžić, and M. Andrejić, "Translating a knowledge base into html," *Knowledge-Based Systems*, vol. 19, no. 1, pp. 92–101, 2006.

[65] G. A. Di Lucca, M. Di Penta, and A. R. Fasolino, "An approach to identify duplicated web pages," in *Computer Software and Applications Conference, 2002. COMPSAC 2002. Proceedings. 26th Annual International*, pp. 481–486, IEEE, 2002.

[66] M. Lučanskỳ, M. Šimko, and M. Bieliková, "Enhancing automatic term recognition algorithms with html tags processing," in *Proceedings of the 12th International Conference on Computer Systems and Technologies*, pp. 173–178, ACM, 2011.

[67] B. A. Kitchenham, L. M. Pickard, and S. J. Linkman, "An evaluation of some design metrics," *Software engineering journal*, vol. 5, no. 1, pp. 50–58, 1990.

[68] J. C. Munson and S. G. Elbaum, "Code churn: A measure for estimating the impact of code change," in *Software Maintenance, 1998. Proceedings., International Conference on*, pp. 24–31, IEEE, 1998.

[69] M. Lučanskỳ, M. Šimko, and M. Bieliková, "Enhancing automatic term recognition algorithms with html tags processing," in *Proceedings of the 12th International Conference on Computer Systems and Technologies*, pp. 173–178, ACM, 2011.

[70] H. Davis, *Search engine optimization*. " O'Reilly Media, Inc.", 2006.

[71] V. Rajlich and N. Wilde, "The role of concepts in program comprehension," in *Proceedings 10th International Workshop on Program Comprehension*, pp. 271–278.

[72] S. Scalabrino, M. Linares-Vásquez, D. Poshyvanyk, and R. Oliveto, "Improving code readability models with textual features," in *2016 IEEE 24th International Conference on Program Comprehension (ICPC)*, pp. 1–10.

[73] D. D. Cowan, D. M. Germán, C. J. P. Lucena, and A. v. Staa, "Enhancing code for readability and comprehension using SGML," in *In International Conference on Software Maintenance*, pp. 181–190, Society Press.

[74] X. Wang, L. Pollock, and K. Vijay-Shanker, "Automatic segmentation of method code into meaningful blocks to improve readability," in *2011 18th Working Conference on Reverse Engineering*, pp. 35–44.

[75] A. D. Lucia, R. Oliveto, F. Zurolo, and M. D. Penta, "Improving comprehensibility of source code via traceability information: a controlled experiment," in *14th IEEE International Conference on Program Comprehension (ICPC'06)*, pp. 317–326.

[76] M. A. G. Gaitani, V. E. Zafeiris, N. A. Diamantidis, and E. A. Giakoumakis, "Automated refactoring to the null object design pattern," vol. 59, pp. 33–52.

[77] B. Carter, "On choosing identifiers," vol. 17, no. 5, pp. 54–59.

[78] K. Nishizono, S. Morisakl, R. Vivanco, and K. Matsumoto, "Source code comprehension strategies and metrics to predict comprehension effort in software maintenance and evolution tasks - an empirical study with industry practitioners," in *2011 27th IEEE International Conference on Software Maintenance (ICSM)*, pp. 473–481.

[79] H. Mössenböck and K. Koskimies, *Active Text for Structuring and Understanding Source Code SOFTWARE-Practice and Experience, 26 (7): 833-850.* July.

[80] R. J. Miara, J. A. Musselman, J. A. Navarro, and B. Shneiderman, "Program indentation and comprehensibility," vol. 26, no. 11, pp. 861–867.

[81] A. A. Bourbonnière, "AN INVESTIGATION lNT0 TEXT COMPREHENSIBILITY IN DYNAMIC ELECTRONIC TUCTS: HYPERTEXT AND HYPERMEDIA."

[82] F. Ricca, E. Pianta, P. Tonella, and C. Girardi, "Improving web site understanding with keyword-based clustering," vol. 20, no. 1, pp. 1–29.

[83] R. Cilibrasi and P. M. B. Vitanyi, "Clustering by compression," vol. 51, no. 4, pp. 1523–1545.

[84] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, *Experimentation in software engineering*. Springer Science & Business Media, 2012.

[85] D. I. Sjoberg, T. Dyba, and M. Jorgensen, "The future of empirical methods in software engineering research," in *2007 Future of Software Engineering*, pp. 358–378, IEEE Computer Society, 2007.

[86] M. Mora, V. Rory, M. Rainsinghani, O. Gelman, *et al.*, "Impacts of electronic process guides by types of user: An experimental study," *International Journal of Information Management*, vol. 36, no. 1, pp. 73–88, 2016.

[87] C. Yoo and G. F. Cooper, "An evaluation of a system that recommends microarray experiments to perform to discover gene-regulation pathways," *Artificial Intelligence in Medicine*, vol. 31, no. 2, pp. 169–182, 2004.

[88] L. Lazic and D. Velasevic, "Applying simulation and design of experiments to the embedded software testing process," *Software Testing Verification and Reliability*, vol. 14, no. 4, pp. 257–282, 2004.

[89] L. A. Notenboom, "Compressing and decompressing text files," Apr. 28 1992. US Patent 5,109,433.

[90] A. G. Gounares, C. M. Franklin, and T. R. Lawrence, "Html/xml tree synchronization," Jan. 20 2004. US Patent 6,681,370.

[91] G. Pant, "Deriving link-context from html tag tree," in *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, pp. 49–55, ACM, 2003.

[92] H. Shahriar and M. Zulkernine, "Mutec: Mutation-based testing of cross site scripting," in *Proceedings of the 2009 ICSE Workshop on Software Engineering for Secure Systems*, pp. 47–53, IEEE Computer Society, 2009.

[93] J. Schimmel, K. Molitorisz, A. Jannesari, and W. F. Tichy, "Combining unit tests for data race detection," in *Automation of Software Test (AST), 2015 IEEE/ACM 10th International Workshop on*, pp. 43–47, IEEE, 2015.

[94] A. Kristensen, "Template resolution in xml/html," *Computer Networks and ISDN Systems*, vol. 30, no. 1, pp. 239–249, 1998.

[95] B. F. Manly, "Randomization and regression methods for testing for associations with geographical, environmental and biological distances between populations," *Researches on Population Ecology*, vol. 28, no. 2, pp. 201–218, 1986.

[96] D. B. Rubin, "Bayesian inference for causal effects: The role of randomization," *The Annals of statistics*, pp. 34–58, 1978.

[97] D. C. Montgomery, *Design and analysis of experiments*. John Wiley & Sons, 2008.

[98] A. L. Brown, "Design experiments: Theoretical and methodological challenges in creating complex interventions in classroom settings," *The journal of the learning sciences*, vol. 2, no. 2, pp. 141–178, 1992.

[99] G. Fraser and A. Arcuri, "Handling test length bloat," *Software Testing, Verification and Reliability*, vol. 23, no. 7, pp. 553–582, 2013.

[100] P. E. Ammann and P. E. Black, "A specification-based coverage metric to evaluate test sets," *International Journal of Reliability, Quality and Safety Engineering*, vol. 8, no. 04, pp. 275–299, 2001.

[101] W. Sack, "Conversation map: a content-based usenet newsgroup browser," in *From Usenet to CoWebs*, pp. 92–109, Springer, 2003.

[102] K. Weber, ", in which pooh proposes improvements to web authoring tools, having seen said tools for the unix platform," *Computer Networks and ISDN Systems*, vol. 27, no. 6, pp. 823–829, 1995.

[103] L. Thabane, J. Ma, R. Chu, J. Cheng, A. Ismaila, L. P. Rios, R. Robson, M. Thabane, L. Giangregorio, and C. H. Goldsmith, "A tutorial on pilot studies: the what, why and how," *BMC medical research methodology*, vol. 10, no. 1, p. 1, 2010.

[104] R. L. Glass, "Pilot studies: What, why and how," *Journal of Systems and Software*, vol. 36, no. 1, pp. 85–97, 1997.

[105] A. C. Leon, L. L. Davis, and H. C. Kraemer, "The role and interpretation of pilot studies in clinical research," *Journal of psychiatric research*, vol. 45, no. 5, pp. 626–629, 2011.

[106] S. S. Wu and M. C. Yang, "Using pilot study information to increase efficiency in clinical trials," *Journal of Statistical Planning and Inference*, vol. 137, no. 7, pp. 2172–2183, 2007.

[107] D. E. Berger, "Introduction to multiple regression," *Claremont Graduate University. Retrieved on Dec*, vol. 5, p. 2011, 2003.

[108] G. K. Uyanık and N. Güler, "A study on multiple linear regression analysis," *Procedia-Social and Behavioral Sciences*, vol. 106, pp. 234–240, 2013.

[109] E. Alexopoulos, "Introduction to multivariate regression analysis," *Hippokratia*, vol. 14, no. Suppl 1, p. 23, 2010.

[110] A. Liang and W. Qihua, "Regression analysis method for software reliability growth test data," in *Proceedings of the 2010 Second World Congress on Software Engineering-Volume 01*, pp. 245–248, IEEE Computer Society, 2010.

[111] L. S. Aiken, S. G. West, and R. R. Reno, *Multiple regression: Testing and interpreting interactions*. Sage, 1991.

[112] J. P. Davim and P. Reis, "Multiple regression analysis (mra) in modelling milling of glass fibre reinforced plastics (gfrp)," *International journal of manufacturing technology and management*, vol. 6, no. 1-2, pp. 185–197, 2004.

[113] S.-M. Huang and J.-F. Yang, "Linear discriminant regression classification for face recognition," *IEEE Signal Processing Letters*, vol. 20, no. 1, pp. 91–94, 2013.

[114] U. Lorenzo-Seva, P. J. Ferrando, and E. Chico, "Two spss programs for interpreting multiple regression results," *Behavior research methods*, vol. 42, no. 1, pp. 29–35, 2010.

[115] D. George and P. Mallery, *IBM SPSS Statistics 23 Step by Step: A Simple Guide and Reference*. Routledge, 2016.

[116] A. Field, *Discovering statistics using IBM SPSS statistics*. Sage, 2013.

[117] M. J. Norušis, *IBM SPSS statistics 19 guide to data analysis*. Prentice Hall Upper Saddle River, New Jersey, 2011.

[118] S. L. Weinberg and S. K. Abramowitz, *Statistics using SPSS: An integrative approach*. Cambridge University Press, 2008.

[119] G. A. Seber and A. J. Lee, *Linear regression analysis*, vol. 936. John Wiley & Sons, 2012.

[120] J. Racine and Q. Li, "Nonparametric estimation of regression functions with both categorical and continuous data," *Journal of Econometrics*, vol. 119, no. 1, pp. 99–130, 2004.

[121] T. Nummi, "Generalised linear models for categorical and continuous limited dependent variables," 2015.

[122] M. I. Coco and R. Dale, "Cross-recurrence quantification analysis of categorical and continuous time series: an r package," *arXiv preprint arXiv:1310.0201*, 2013.

[123] A. Agresti and I. Liu, "Strategies for modeling a categorical variable allowing multiple category choices," *Sociological Methods & Research*, vol. 29, no. 4, pp. 403–434, 2001.

[124] I.-G. Chong and C.-H. Jun, "Performance of some variable selection methods when multicollinearity is present," 2005.

[125] M. H. Graham, "Confronting multicollinearity in ecological multiple regression," *Ecology*, vol. 84, no. 11, pp. 2809–2815, 2003.

[126] J. M. Cortina, "Interaction, nonlinearity, and multicollinearity: Implications for multiple regression," *Journal of Management*, vol. 19, no. 4, pp. 915–922, 1993.

[127] B. Scholkopft and K.-R. Mullert, "Fisher discriminant analysis with kernels," *Neural networks for signal processing IX*, vol. 1, no. 1, p. 1, 1999.

[128] P. Xanthopoulos, P. M. Pardalos, and T. B. Trafalis, "Linear discriminant analysis," in *Robust Data Mining*, pp. 27–33, Springer, 2013.

[129] S. Balakrishnama and A. Ganapathiraju, "Linear discriminant analysis-a brief tutorial," *Institute for Signal and information Processing*, vol. 18, 1998.

[130] J. Zhao, L. Philip, L. Shi, and S. Li, "Separable linear discriminant analysis," *Computational Statistics & Data Analysis*, vol. 56, no. 12, pp. 4290–4300, 2012.

[131] W. F. Lavelle, K. Albanese, N. R. Ordway, and S. A. Albanese, "A stepwise multiple regression analysis of pedicle screws in the thoracolumbar spine," *The Spine Journal*, vol. 14, no. 11, p. S157, 2014.

[132] L. Li, "Quantifying tio2 abundance of lunar soils: Partial least squares and stepwise multiple regression analysis for determining causal effect," *Journal of Earth Science*, vol. 22, no. 5, pp. 549–565, 2011.

[133] Y. Zhang, H. Ma, B. Wang, W. Qu, A. Wali, and C. Zhou, "Relationships between the structure of wheat gluten and ace inhibitory activity of hydrolysate: stepwise multiple linear regression analysis," *Journal of the Science of Food and Agriculture*, 2015.

[134] A. Kolasa-Wiecek, "Stepwise multiple regression method of greenhouse gas emission modeling in the energy sector in poland," *Journal of Environmental Sciences*, vol. 30, pp. 47–54, 2015.

[135] R. Feldt and A. Magazinius, "Validity threats in empirical software engineering research-an initial survey.," in *SEKE*, pp. 374–379, 2010.

[136] M. Daun, A. Salmon, T. Bandyszak, and T. Weyer, "Common threats and mitigation strategies in requirements engineering experiments with student participants," in *International Working Conference on Requirements Engineering: Foundation for Software Quality*, pp. 269–285, Springer, 2016.

[137] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, *Experimentation in software engineering*. Springer Science & Business Media, 2012.

Appendices

Appendix A

Metrics related to test input

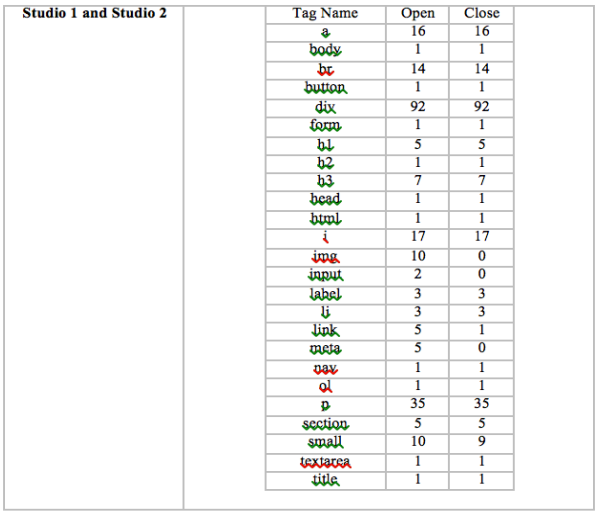
The tags detailed classification based on each tag that is used throughout each and every HTML test input is presented below:

Blue Media 1 and Blue media 2		Tag Name	Open	Close	
		a	33	33	
		body	1	1	
		br	4	4	
		div	47	47	
		form	1	1	
		h1	6	6	
		h2	1	1	
		head	1	1	
		html	1	1	
		img	8	8	
		input	2	2	
		li	16	16	
		link	1	1	
		meta	3	3	
		p	8	8	
		span	6	6	
		title	1	1	
		ul	3	3	
Blue Simple Template		Tag Name	Open	Close	
		a	59	59	
		abbr	1	1	
		acronym	1	1	
		blockquote	1	1	
		body	1	1	
		br	2	2	
		button	1	1	
		cite	1	1	
		div	63	63	
		form	1	1	
		h1	2	2	
		h2	4	4	
		h3	5	5	
		h4	1	1	
		h5	1	1	
		h6	1	1	
		head	1	1	
		html	1	1	
		img	3	3	
		input	2	2	
		li	35	35	
		link	1	1	
		meta	1	1	
		p	8	8	
		span	12	12	
		sub	2	2	
		sup	2	2	
		table	3	3	
		tbody	2	2	
		td	14	14	
		th	3	3	

		title	1	1	
		tr	6	6	
		ul	11	11	
Lady tulip		Tag Name	Open	Close	
		a	26	26	
		body	1	1	
		div	29	29	
		h1	1	1	
		h2	5	5	
		h3	4	4	
		head	1	1	
		html	1	1	
		img	4	4	
		li	11	11	
		link	3	3	
		meta	3	3	
		p	11	11	
		span	10	10	
		strong	1	1	
		title	1	1	
		ul	2	2	
Cooperation		Tag Name	Open	Close	
		a	24	24	
		body	1	1	
		br	2	2	
		div	26	26	
		fieldset	2	2	
		form	2	2	
		h1	2	2	
		h2	7	7	
		head	1	1	
		html	1	1	
		img	7	7	
		input	8	8	
		label	5	5	
		legend	2	2	
		li	27	27	
		link	1	1	
		meta	1	1	
		p	24	24	
		span	3	3	
		strong	3	3	
		textarea	1	1	
		title	1	1	
		ul	7	7	
Escape velocity 1 and escape velocity 2		Tag Name	Open	Close	
		a	36	36	
		body	1	1	
		br	6	6	
		div	46	46	
		form	1	1	

		b1	1	1	
		b2	2	2	
		b3	13	13	
		head	1	1	
		header	2	2	
		br	2	2	
		html	1	1	
		img	4	4	
		input	4	4	
		li	25	25	
		link	1	1	
		meta	2	2	
		nav	1	1	
		p	19	19	
		section	17	17	
		strong	3	3	
		textarea	1	1	
		title	1	1	
		ul	10	10	
Forty		Tag Name	Open	Close	
		a	23	23	
		article	6	6	
		body	1	1	
		br	3	3	
		div	13	13	
		footer	1	1	
		form	1	1	
		h1	1	1	
		h2	1	1	
		h3	9	9	
		head	1	1	
		header	9	9	
		html	1	1	
		img	6	6	
		input	4	4	
		label	3	3	
		li	17	17	
		link	1	1	
		meta	2	2	
		nav	2	2	
		p	8	8	
		section	9	9	
		span	17	17	
		strong	1	1	
		textarea	1	1	
		title	1	1	
		ul	7	7	
Intensify 1 and intensify 2		Tag Name	Open	Close	
		a	12	12	
		blockquote	1	1	
		body	1	1	

		br	6	6	
		cite	2	2	
		div	22	22	
		figure	1	1	
		footer	2	2	
		h1	1	1	
		h2	2	2	
		h3	7	7	
		head	1	1	
		header	1	1	
		html	1	1	
		img	4	4	
		li	8	8	
		link	1	1	
		meta	2	2	
		nav	3	3	
		p	8	8	
		section	4	4	
		span	4	4	
		title	1	1	
		ul	4	4	
Coefficient 1 and coefficient 2		Tag Name	Open	Close	
		a	22	22	
		body	1	1	
		div	27	27	
		h1	1	1	
		h2	5	5	
		h3	4	4	
		head	1	1	
		html	1	1	
		img	3	3	
		li	11	11	
		link	3	3	
		meta	3	3	
		p	10	10	
		span	6	6	
		strong	1	1	
		title	1	1	
		ul	2	2	



Studio 1 and Studio 2

Tag Name	Open	Close
a	16	16
body	1	1
br	14	14
button	1	1
div	92	92
form	1	1
h1	5	5
h2	1	1
h3	7	7
head	1	1
html	1	1
i	17	17
img	10	0
input	2	0
label	3	3
li	3	3
link	5	1
meta	5	0
nav	1	1
ol	1	1
p	35	35
section	5	5
small	10	9
textarea	1	1
title	1	1

Figure A.1: Different tags that are applied on each HTML test input that this study has selected are clearly illustrated.

Appendix B

Pre-Questionnaire and Post-Questionnaire

Pre-Questionnaire for Pilot Study 1:

1. Email address
2. Name
3. Specialization
4. Do you have any knowledge in HTML?
5. Select your knowledge level in HTML?
6. What are your available timings?

Pre-Questionnaire for Pilot Study 2:

1. Name
2. Email Id
3. Which group are you from?
4. Do you have knowledge in HTML?
   (a) Yes
   (b) No
5. Select your knowledge level in HTML?

Post-Questionnaire for Pilot study 1 and 2:

Thank you text:

We thank you for your participation in the experiment. We would like to take this opportunity to thank our supervisor Dr. Simon poulding for supporting us and help to achieve this target and get the data we from you.

We request you to participate in the post questionnaire given to you and let us know the feedback and the type of experience you gained from the experiment also mention the difficulties in answering the questions.

1. Name
2. Email
3. What are the challenges you faced while doing the experiment?

   (a) Difficulties in understanding the code
   (b) Experiment setup
   (c) Time constraints
   (d) External factors like environment
   (e) Other

4. Describe your experience?

   (a) Excellent
   (b) Very Good
   (c) Good
   (d) Fair poor

5. Any recommendations?

Pre-Questionnaire for the Experiment:

1. Email address
2. Name
3. Which course are you taking?
4. Do you have intermediate/expert level in HTML?

Appendix C

Experiment Invitation

C.1 Cover letter for Master Thesis Students:

Invitation to participate in the research experiment titled: "*Automatically generating Realistic and Comprehensible Test data*"

Dear Student,
We are conducting an experiment for a research study focused on improving the understanding on metrics that are used on the test data, those that are cost effective and are good predictors of human oracle costs. As a technical person (developer/tester) you are in an ideal position to give us valuable first-hand information from your own perspective. Having good knowledge (intermediate/expert level) in HTML is the only requirement for you to be able to participate in this experiment.

The experiment takes around 75 minutes on **9th and 10th of December** at lab **H322, BTH**. If you are willing to participate in the experiment, please sign up in the following links provided as per your available date and time:

To participate on 9th Dec, Fri 10:00:

To participate on 9th Dec, Fri 13:00:

To participate on 10th Dec, Sat 10:00:

To participate on 10th Dec, Sat 13:00:

By signing up, you are indicating that you intend to be present at the experiment. A remainder mail will also be sent to your provided email address one day before the experiment. We recommend you to contact us through mail if you sign up but you could not attend in any case.

Looking forward to meet you.

Thank you,
Kavya Chelluboina
E-mail: ch.kavya2009@gmail.com
Kartheek Chilla
E-mail: chilla.kartheek87@gmail.com

Figure C.1: Cover letter for Master Thesis Students

C.2 Cover letter for Vinnova students:

Invitation to participate in the research experiment titled: "*Automatically generating Realistic and Comprehensible Test data*"

Dear Student,
We are conducting an experiment for a research study focused on improving the understanding on metrics that are used on the test data, those that are cost effective and are good predictors of human oracle costs. As a technical person (developer/tester) you are in an ideal position to give us valuable first-hand information from your own perspective. Having good knowledge (intermediate/expert level) in HTML is the only requirement for you to be able to participate in this experiment.

The experiment takes around 75 minutes on **10th of December** at lab **H322, BTH**. If you are willing to participate in the experiment, please sign up in the following links provided as per your available time:

To participate on 10th Dec, Sat 10:00:

To participate on 10th Dec, Sat 13:00:

By signing up, you are indicating that you intend to be present at the experiment. A remainder mail will also be sent to your provided email address one day before the experiment. We recommend you to contact us through mail if you sign up but you could not attend in any case.

Looking forward to meet you.

Thank you,
Kavya Chelluboina
E-mail: ch.kavya2009@gmail.com
Kartheek Chilla
E-mail: chilla.kartheek87@gmail.com

Figure C.2: Cover letter for Master Thesis Students

C.3 Mail sent to the participants for the experiment:


Hello,

WE REQUEST YOU TO PLEASE OPEN THE LINK GIVEN BELOW:

http://experimentbthmasterthes is.limequery.com/survey/index/sid/415988/newtest/Y/lang/en

Thank you!

Figure C.3: Cover letter for Master Thesis Students

C.4 During Presentation:

Conversation used during presentation just before the pilot study and experiment starts

- We want to look at what makes the test data effective so we want You to look into the test input and see if the output displayed is correct or not.
- They are in random order with a time limit. You should make sure that you answer every question and also should answer the given questions in the same order.
- Answer the questions accurately it doesn't matter how far you get. It doesn't matter how many you get done the more important thing is to try get the answers correctly.
- For every question you need to give your confidence level between the reciter scale which is not available so you can give it for example five by ten. For the answer doesn't know in case if it is selected, you should comment it for example: hard OR very hard OR input/output not clear.
- As soon as the 60 minutes' time is finished you should stop answering the questions and don't even have to guess the questions.

Appendix D

Test Input Selection

The Table given below describes which test inputs are used in which Pilot studies and Experiments.

Test Input	Output displayed	ID	Pilot Study 1	Pilot Study 2	Experiment 1	Final Experiment
Art gallery 1	Wrong	ID1	Yes		Yes	Yes
Art gallery 2	Wrong	ID2		Yes		
Aerial 1	Wrong	ID3	Yes		Yes	Yes
Aerial 2	Wrong	ID4		Yes		
Black Coffee	Wrong	ID5		Yes		
Lady Tulip	Correct	ID6	Yes		Yes	Yes
Blue Media 1	Wrong	ID7	Yes		Yes	Yes
Blue media 2	Wrong	ID8		Yes		
Blue simple template	Wrong	ID9	Yes		Yes	Yes
Cooperation	Wrong	ID10		Yes		
Escape Velocity 1	Correct	ID11	Yes		Yes	Yes
Escape Velocity 2	Wrong	ID12		Yes		
Forty	Wrong	ID13	Yes		Yes	Yes
Intensify 2	Wrong	ID14		Yes		
Studio 1	Correct	ID15		Yes		
Studio 2	Wrong	ID16	Yes		Yes	Yes
Coefficient 1	Wrong	ID17	Yes		Yes	Yes
Coefficient 2	Correct	ID18		Yes		
Intensify 1	Correct	ID19	Yes		Yes	Yes

Figure D.1: Different test inputs used in pilot study 1, Pilot study2 and the experiments.

Test Input Questions

The HTML test inputs are addressed below in the form of links which direct them to the PDF file. It is hard to include all the test input in the word document so, we generated a link for every question and shared them in the Google drive.

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16

- 17
- 18
- 19

Image for Time statistics for each test input:

Time statistics

	ID	Total time	Group: TEST INPUT 1	Question: Q1Time	Group: TEST INPUT 2	Question: Q3Time	Group: TEST INPUT 3	Question: Q6Time	Group: TEST INOUT 4	Question: Q7Time	Group: TEST INPUT 5	Question: Q9Time	Group: TEST INPUT 6	Question: Q11Time	Group: TEST INPUT 7	Qu Q1
	1	3255.83	334.55		37.49		431.4		371.94		141.68		535.17		392.16	
	2	2677.94	137.26		95.23		188.15		466.06		389.71		332.84		171.18	
	3	3035.29	40.86		75.04		434.73		383.52		648.61		492.4		55.35	
	4	2722.43	238.39		207.64		326.02		374.04		276.81		243.74		250.61	
	5	2440.05	113.7		80.55		104.92		209.23		171.13		389.77		364.08	
	6	3038.59	99		208.6		82.88		96.92		181.92		305.64		680.07	
	7	2362.14	481.04		113.93		158.67		372.83		147.2		231.98		201.32	
	8	0														
	9	3401.38	272.95		356.73		391.86		271.35		169		540.4		184.93	
	11	0														

« ‹ 1 2 3 4 › »

Figure D.2: Time taken by each participant to answer each test input is gathered from Lime Survey storage statistics.

SPSS Statistics Variable Table:

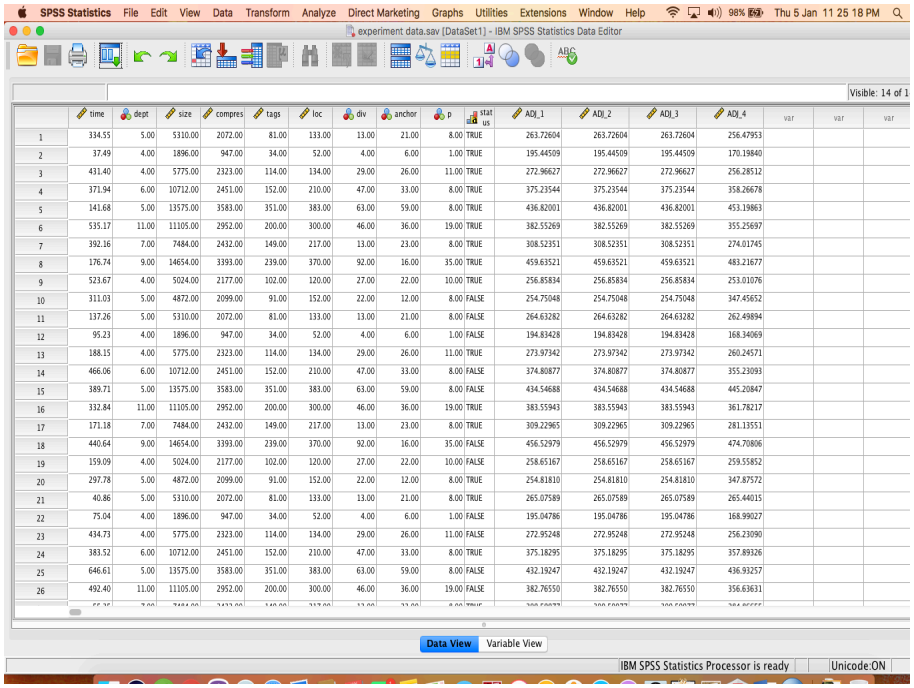


Figure D.3: Statistics about the correct or wrong answer mentioned by the participants.

Appendix E

Results from Pilot Study 1 and 2, and Experiment

E.1 Pilot study 1 graphs and results:

Test Input	Question ID	Depth	Tags	Size (bytes)	Compress size (bytes)
1	1	4	81	5310	2072
2	3	4	34	1896	947
3	6	4	114	5775	2323
4	7	6	152	10712	2451
5	9	5	351	13575	3583
6	11	11	200	11105	2952
7	13	7	149	7484	2432
8	16	9	239	14654	3393
9	17	4	102	5024	2177
10	19	5	91	4872	2099

Table E.1: The selected test inputs for the Pilot study 1 and their corresponding ID's and all the four metrics variation are illustrated.

Below we provide a brief analysis for the pilot study although the analysis is not important for the study as the number of participant's attempt are less we provide this as a measure to understand and validate if all the required data that is necessary to perform actual analysis is gathered without any problems. Thus we performed a small brief analysis in both pilots.

Regression analysis: Correlations table helps to understand how different table interact with each other [66]. If the dependent variable and independent variable are highly correlated to each other then they are multi-collinearity. From the Pearson correlations row in the correlations diagram we observe that the tags have positive correlation with time unlike other variables. For instance, our diagram shows that the time has negative correlation with depth size and compress size and positive correlation with the number of tags that is 1.0000 and 0.232. ANOVA section helps to understand the variance in the statistical model have degree of freedom df =k that indicates how many regressors the model has so k=4 so we have 4 regressors. Total number of observations is 40 so N =16. Then total degree of freedom is N-1=39 and the total residual is n-k-1= 35. Regression helps to find the variability. We have sum of squares of regression (SSR) which is 0.973,

sum of squares total (SST) which is 3.314 and finally sum of square of residual (SSR) which is 2.341. significance of total regression is 0.14. The ANNOVA table presents the statistic f-test =3.637. the p value is <0.05 which means the model is statistically significant. the equation built from the coefficient table is as follows

[DF]

[R1] y= b0+b1x1+b2x2+b3x3+b4x4 which is

y= 0.690+(-16.065) (Depth)+0.10 (Size)+(-0.244) (compress size) +0.964 (number of tags).

Analysis Summary: In the model summary table, the r-squared value is given to understand how much variation is there among variables by multiplying the r-squared value with 100 to obtain the percentage that is 0.294*100= 29.4% this is 29.4% of the variance in time can be accounted for by the predictors size, compress size, depth and number of tags. The adjusted r-squared value in table is used when we are using a small sample size. To calculate this, combine related f value to the r square value. Represented r2 (r-squared) = .294 (r-squared), F (4, 35) (regression degrees of freedom, residual degree of freedom) = 3.637 (F value), (statistical significance) p=0.14 (significant).

From the standardized coefficients beta, we can compare the variables beta level here Standardized means for each of these values that are in the column the value of the variables is converted to same scale so that it is easy to compare them. Compress size with .602 is the largest value among all the variables present, the compress size thus makes the strongest contribution to the outcome when variance is explained by all the other variables in the model. The statistical significance of the model, which helps to note the statistically significant contribution of individual variable to the prediction model and this is dependent on which variables that are included in the equation and how much overlap/collinearity is present among these independent variables. If the significant value is less than 0.05 then the model makes a significant contribution and the value is >0.05 then it is not making a significant contribution of individual variable to the prediction model. Similarly, a large t-value paired with small significance value suggest that the predictor value have a large impact on criterion variable. Beta = beta value (0.491), t= t-value (3.006), and p =0.005 (significant). The tags have a positive coefficient of 0.964 which means for every 1 unit increase in the number of tags the time increases by 0.964 seconds. So, to summarize the results form pilot study 1 the analysis allows to answer a couple of questions like the 29.4 % of variance in the time taken to answer. The number of tags makes largest unique statistically significant unique contribution to the outcome among all the variable present.

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.542[a]	.294	.213	.25862196

a. Predictors: (Constant), tags, depth, compress, size
b. Dependent Variable: time

ANOVA[a]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	.973	4	.243	3.637	.014[b]
	Residual	2.341	35	.067		
	Total	3.314	39			

a. Dependent Variable: time
b. Predictors: (Constant), tags, depth, compress, size

Descriptive Statistics

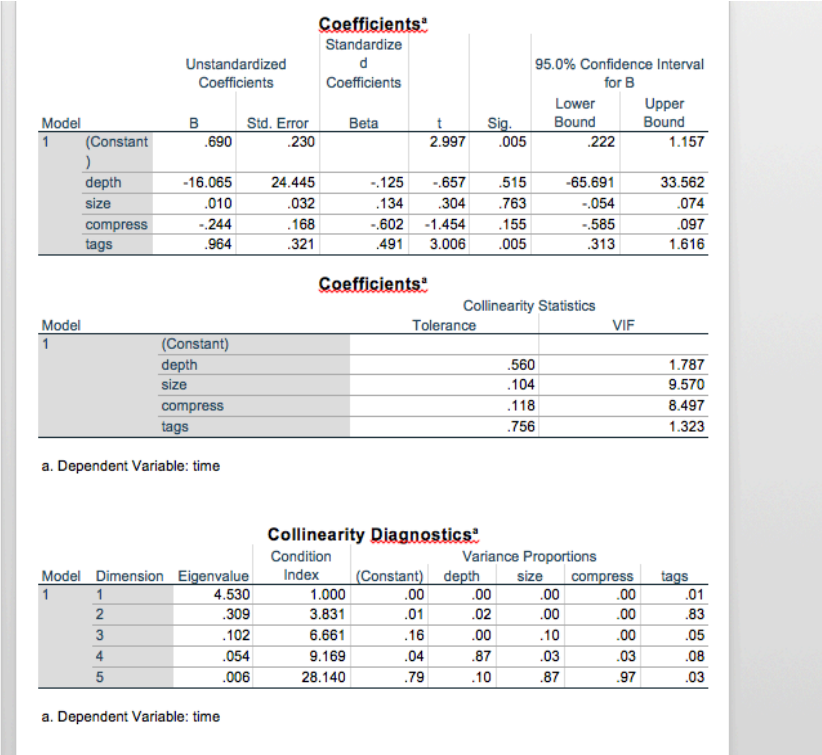
	Mean	Std. Deviation	N
time	.2394925	.29151089	40
depth	.00600	.002265	40
size	8.04070	4.059664	40
compress	2.44290	.719010	40
tags	.17178	.148467	40

Figure E.1: Model Summary, ANOVA and Descriptive Statistics for Pilot study 1

Correlations

		time	depth	size	compress	tags
Pearson Correlation	time	1.000	-.274	-.298	-.308	.232
	depth	-.274	1.000	.645	.549	.195
	size	-.298	.645	1.000	.933	.430
	compress	-.308	.549	.933	1.000	.485
	tags	.232	.195	.430	.485	1.000
Sig. (1-tailed)	time	.	.044	.031	.027	.075
	depth	.044	.	.000	.000	.114
	size	.031	.000	.	.000	.003
	compress	.027	.000	.000	.	.001
	tags	.075	.114	.003	.001	.
N	time	40	40	40	40	40
	depth	40	40	40	40	40
	size	40	40	40	40	40
	compress	40	40	40	40	40
	tags	40	40	40	40	40

Figure E.2: Correlations among metric independent and time dependent for Pilot study 1



Coefficients[a]

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.	95.0% Confidence Interval for B Lower Bound	95.0% Confidence Interval for B Upper Bound
1	(Constant)	.690	.230		2.997	.005	.222	1.157
	depth	-16.065	24.445	-.125	-.657	.515	-65.691	33.562
	size	.010	.032	.134	.304	.763	-.054	.074
	compress	-.244	.168	-.602	-1.454	.155	-.585	.097
	tags	.964	.321	.491	3.006	.005	.313	1.616

Coefficients[a]

Model		Collinearity Statistics Tolerance	Collinearity Statistics VIF
1	(Constant)		
	depth	.560	1.787
	size	.104	9.570
	compress	.118	8.497
	tags	.756	1.323

a. Dependent Variable: time

Collinearity Diagnostics[a]

Model	Dimension	Eigenvalue	Condition Index	Variance Proportions (Constant)	depth	size	compress	tags
1	1	4.530	1.000	.00	.00	.00	.00	.01
	2	.309	3.831	.01	.02	.00	.00	.83
	3	.102	6.661	.16	.00	.10	.00	.05
	4	.054	9.169	.04	.87	.03	.03	.08
	5	.006	28.140	.79	.10	.87	.97	.03

a. Dependent Variable: time

Figure E.3: Coefficients and collinearity statistics for Pilot study1

E.2 Pilot study 2:

The selected test inputs for the Pilot study 2 and their corresponding ID's and all the four metrics variation are illustrated.

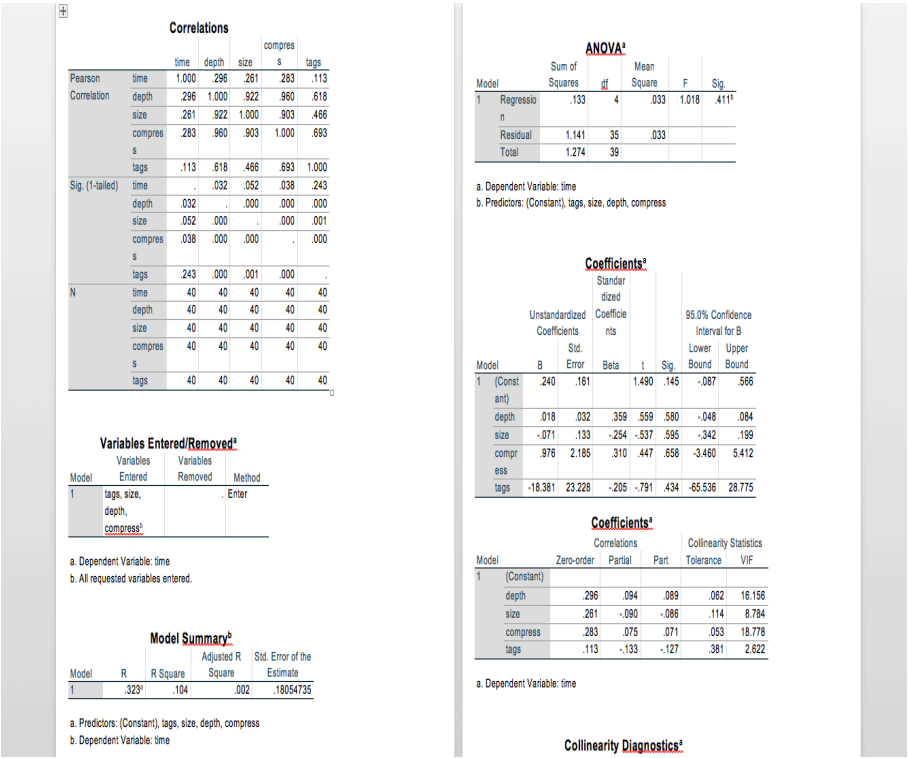
Test Input	Question ID	Depth	Tags	Size (bytes)	Compress size (bytes)
1	2	4	81	5310	2072
2	4	4	34	1905	931
3	5	8	124	5921	1695
4	8	5	143	9856	2247
5	10	5	158	8532	2576
6	12	9	200	10186	2513
7	14	3	99	4464	1647
8	15	7	239	14654	3386
9	18	4	102	5024	2237
10	19	3	99	4464	1647

Table E.2: The selected test inputs for the Pilot study 2 and their corresponding ID's and all the four metrics variation are illustrated.

Regression analysis: To understand how much variance is present in the model summary table, the r-squared value is helpful. In ANOVA we have total three columns in Model figure namely regression, residual and total. To understand variance, the ANOVA

model can be helpful. To know how much of the variance in time is perceived by the independent variables is deduced form the Summary model table. or a good prediction the study should have enough variability or variance. To find out the variance the regression analysis is helpful. The relationship between independent and dependent variables can be identified form the correlations table. If the multi collinearity exist among the variables in the multiple regression it reduces the accuracy of the model [R1]. Pearson coalitions helps to identify when one factor goes up then the other factor goes up as well. This helps to identify coalitions among the factors using Pearson coalitions. The correlation table help to understand how different variables interact with each other [R1]. Form the table if observed all the independent variable have positive relationship with the dependent variable time. For standardized coefficients beta, we can compare variables beta level the standardized in the sense all the variables are converted to same scale this makes the comparison easy. To understand the whether the model is significant or not is possible through the F-value for which the p-value should be below 0.05 but in this case P>0.05. If the value is <0.05 then it makes a significant contribution to the model. In the table there is no particular independent variable that is <0.05 so no individual model is making significant individual contribution to the outcomes.

Model summary: The values are recorded in percentage so after multiplying with value 0.104 *100 =10.4% of variance in time is accounted by the predictors size, compress size, depth and tags that is 10.4% of total variability in dependent variable is explained by independent variable. The related effort t-value helps to estimate the overall significance of the model. To calculate this, combine the related f value with the r-square value. Represented r-squared=0.104, F (4,35) = 1.018 (f value); statistical significance p=0.411. from the collinearly statistics in the coefficients table the tolerance and VIF columns are noticeable. Values with tolerance <0.1 indicates high multi collinearity. The depth and compress size have values 0.089 <0.1 and 0.071 <0.1 which indicates they have high multi collinearity similarly the value of Variation indication factor VIF for depth and compress size are above 10 that is 16.156 >10 and 18.778>10 which indicates they are multi collinearity. Depth with value of 0.359 which is highest among all the four variables which indicates that it makes strongest contribution it the outcome. The order of contribution to the outcome is as follows depth, compress size, number of tags followed by size. The degree of freedom df =k for the pilot study 2 which indicates number of regressors the model has I k=4. Total degree of freedom =N-1=39. And total residual for pilot study =N-k-1=35. The regression helps to find out the variability of the model. The significance of total regression is 0.411. similarly, other important data like sum of square of regression (0.133); sum of square of residual (1.141) are given in ANOVA table. From the Pearson correlation column in the correlation table it is clear that all the four metrics have positive relationship with time. Among them depth is highly correlating with time (0.296) followed by compress size (0.283) then size (0.261) and at last the number of tags (0.113) is positively correlating but considerably small when compared to depth. From the table the outcomes measurement in terms of depth and compress size are as follows for 1 unit increase in the independent variable depth the corresponding time increase by 0.24 seconds and moreover for 1 unit increase in the compress size variable the time increase by 0.976 seconds. So, to summarize from the pilot study 2 helps to understand variance in the time.



Correlations

		time	depth	size	compress	tags
Pearson Correlation	time	1.000	.296	.261	.283	.113
	depth	.296	1.000	.922	.960	.618
	size	.261	.922	1.000	.903	.466
	compress	.283	.960	.903	1.000	.693
	tags	.113	.618	.466	.693	1.000
Sig. (1-tailed)	time	.	.032	.052	.038	.243
	depth	.032	.	.000	.000	.000
	size	.052	.000	.	.000	.001
	compress	.038	.000	.000	.	.000
	tags	.243	.000	.001	.000	.
N	time	40	40	40	40	40
	depth	40	40	40	40	40
	size	40	40	40	40	40
	compress	40	40	40	40	40
	tags	40	40	40	40	40

Variables Entered/Removed[a]

Model	Variables Entered	Variables Removed	Method
1	tags, size, depth, compress[b]		. Enter

a. Dependent Variable: time
b. All requested variables entered.

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.323[a]	.104	.002	.18054735

a. Predictors: (Constant), tags, size, depth, compress
b. Dependent Variable: time

ANOVA[a]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	.133	4	.033	1.018	.411[b]
	Residual	1.141	35	.033		
	Total	1.274	39			

a. Dependent Variable: time
b. Predictors: (Constant), tags, size, depth, compress

Coefficients[a]

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.	95.0% Confidence Interval for B Lower Bound	95.0% Confidence Interval for B Upper Bound
1	(Constant)	.240	.161		1.490	.145	-.087	.566
	depth	.018	.032	.359	.559	.580	-.048	.084
	size	-.071	.133	-.254	-.537	.595	-.342	.199
	compress	.976	2.185	.310	.447	.658	-3.460	5.412
	tags	-18.381	23.228	-.205	-.791	.434	-65.536	28.775

Coefficients[a]

Model		Correlations Zero-order	Correlations Partial	Correlations Part	Collinearity Statistics Tolerance	Collinearity Statistics VIF
1	(Constant)					
	depth	.296	.094	.089	.062	16.156
	size	.261	-.090	-.086	.114	8.784
	compress	.283	.075	.071	.053	18.776
	tags	.113	-.133	-.127	.381	2.622

a. Dependent Variable: time

Collinearity Diagnostics[a]

Figure E.4: The correlations, Model Summary, ANOVA, Coefficients results generated for Pilot Study 2

Time vs metric	**Linear regression equation Y =b0+b1X1**	**R square**	**Standard coefficient B beta level**	**T-value**	**Significance p value**
Depth	189.621+21.771(depth)	0.037	.193	3.516	0.000502*
Size	154.773+0.021(size)	0.105	.328	6.188	1.8796E-9*
Compress size	41.036+.114(compress)	0.104	.322	6.074	3.5492E-9*
Tags	188.328+.872(tags)	0.091	.302	5.654	3.482E-8*
Loc	158.850+.779(loc)	0.107	.327	6.177	1.9904E-9*
Div	206.755+3.188(div)	0.104	.323	6.082	3.4121E-9*
anchor	235.453+3.338(anchor)	0.035	.187	3.389	0.000791*
p	240.784+6.850(p)	0.058	.241	4.424	0.000013*

Table E.3: The Linear regression equation for Time vs 1 metric independent variable and Significance values are illustrated.

ID	**Time vs 2 metric combination**	**F value**	**Significance P value**	**T value**	**Significance**
1	Depth and Size	19.194	1.3542E-8	Depth -.445 Size .348	Depth .657 size .000*
2	Depth and Compress	18.466	2.5936E-8	Depth .366 Compress 4.867	Depth .714 Compress .000*
3	Depth and Number of tags	16.873	1.0878E-7	Depth 1.305 Tags 4.541	Depth .193 Tags .000*
4	Depth and Lines of code	19.171	1.3817E-8	Depth -5.19 LOC 5.005	Depth .604 LOC .000*
5	Depth and Div tag	18.578	2.3464E-8	Depth .508 Div 4.889	Depth .612 Div .000*
6	Depth and anchor tag	10.512	0.000038	Depth 3.040 Anchor 2.894	Depth .003* Anchor .004*
7	Depth and <p>	9.944	0.000065	Depth .589 P 2.698	Depth .550 P . 007*
8	Size and Compress	19.509	1.0227E-8	Size 1.415 Compress .872	Size .158 Compress .384
9	Size and Number of tags	19.107	1.4623E-8	Size 2.401 Tags .210	Size .017* Tags .834

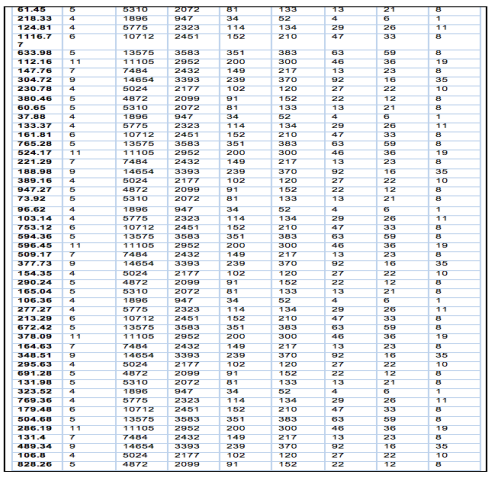
10	Size and Lines of code	19.414	1.1124E-8	Size .839 LOC .769	Size .402 LOC .442
11	Size and div tag	19.670	8.861E-9	Size 1.488 Div 1.024	Size .138 Div .307
12	Size and anchor tag	19.258	1.2789E-8	Size 5.111 Anchor -5.59	Size .000* Anchor .577
13	Size and <p>	19.186	1.3632E-8	Size 4.216 P .430	Size .000* P .668
14	Compress size and number of tags	18.401	2.7502E-8	Compress 2.116 Tags .130	Compress .035* Tags .896
15	Compress size and lines of code	19.290	1.2427E-8	Compress .694 LOC 1.269	Compress .488 LOC .205
16	Compress size and div tag	20.089	6.1074E-9	Compress 1.722 Div 1.744	Compress .086 Div .082
17	Compress size and anchor tag	19.107	1.4623E-8	Compress 5.083 Anchor -1.133	Compress .000* Anchor .258
18	Compress size and <p>	18.732	2.0609E-8	Compress 4.111 p .771	Compress .000* P .441
19	Number of tags and lines of code	19.185	1.3648E-8	Tag -.542 LOC 2.430	Tag .588 LOC .016*
20	Number of tags and div	19.511	1.0206E-8	Tags 1.389 Div 2.549	Tags .166 Div .011*
21	Number of tags and anchor	17.860	4.4702E-8	Tags 4.840 Anchor -1.870	Tags .000* P .030*
22	Number of tags and <p>	18.551	2.4053E-8	Tags 4.071 P 2.180	LOC .072 Anchor .139
23	Lines of code and anchor	20.251	5.2882E-9	LOC 1.804 Anchor 1.482	LOC .000* Div .511
24	Lines of code and div	19.263	1.2727E-8	LOC 5.112 Div -658	LOC .000* P .505
25	Lines of code and <p>	19.270	1.2653E-8	LOC 4.234 P .667	LOC .000* P .505
26	Anchor and div	18.954	1.677E-8	Anchor 5.054 Div .965	Anchor 000* Div .335
27	Anchor and <p>	36.985	3.4121E-9	Anchor 3.514 P 4.520	Anchor .001* P .000*
28	Div and <p>	16.307	1.8158E-7	Div 4.097 P -.642	Div .000* P .522

Table E.4: The Time vs 2 metric independent variable with corresponding t values and Significance values are illustrated.

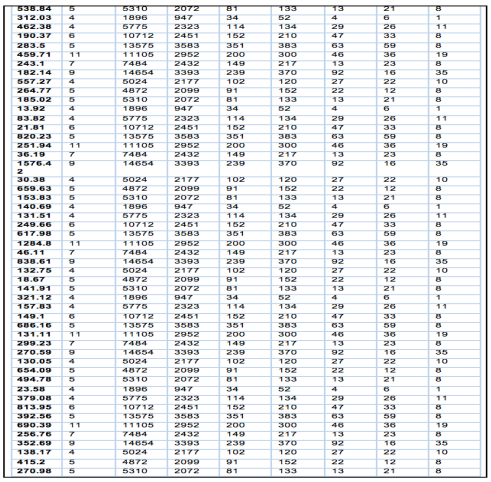
E.3 Final Experiment Results:

Time	depth	size	compress	tags	LOC	<Div>	Anchor tag	<p>
334.55	5	5310	2072	81	133	13	21	8
37.49	4	1896	947	34	52	4	6	1
431.4	4	5775	2323	114	134	29	26	11
371.94	6	10712	2451	152	210	47	33	8
141.68	5	13575	3583	351	383	63	59	8
535.17	11	11105	2952	200	300	46	36	19
392.16	7	7484	2432	149	217	13	23	8
176.74	9	14654	3393	239	370	92	16	35
523.67	4	5024	2177	102	120	27	22	10
311.03	5	4872	2099	91	152	22	12	8
137.26	5	5310	2072	81	133	13	21	8
95.23	4	1896	947	34	52	4	6	1
188.15	4	5775	2323	114	134	29	26	11
466.06	6	10712	2451	152	210	47	33	8
389.71	5	13575	3583	351	383	63	59	8
332.84	11	11105	2952	200	300	46	36	19
171.18	7	7484	2432	149	217	13	23	8
440.64	9	14654	3393	239	370	92	16	35
159.09	4	5024	2177	102	120	27	22	10
297.78	5	4872	2099	91	152	22	12	8
40.86	5	5310	2072	81	133	13	21	8
75.04	4	1896	947	34	52	4	6	1
434.73	4	5775	2323	114	134	29	26	11
383.52	6	10712	2451	152	210	47	33	8
646.61	5	13575	3583	351	383	63	59	8
492.4	11	11105	2952	200	300	46	36	19
55.35	7	7484	2432	149	217	13	23	8
469.3	9	14654	3393	239	370	92	16	35
264.8	4	5024	2177	102	120	27	22	10
170.68	5	4872	2099	91	152	22	12	8
238.39	5	5310	2072	81	133	13	21	8
207.64	4	1896	947	34	52	4	6	1
326.02	4	5775	2323	114	134	29	26	11
374.04	6	10712	2451	152	210	47	33	8
276.81	5	13575	3583	351	383	63	59	8
243.74	11	11105	2952	200	300	46	36	19
250.61	7	7484	2432	149	217	13	23	8
547.09	9	14654	3393	239	370	92	16	35
95.38	4	5024	2177	102	120	27	22	10
162.71	5	4872	2099	91	152	22	12	8
113.7	5	5310	2072	81	133	13	21	8
80.55	4	1896	947	34	52	4	6	1
104.92	4	5775	2323	114	134	29	26	11
209.23	6	10712	2451	152	210	47	33	8
171.13	5	13575	3583	351	383	63	59	8
389.77	11	11105	2952	200	300	46	36	19
364.08	7	7484	2432	149	217	13	23	8
571.83	9	14654	3393	239	370	92	16	35
295.55	4	5024	2177	102	120	27	22	10

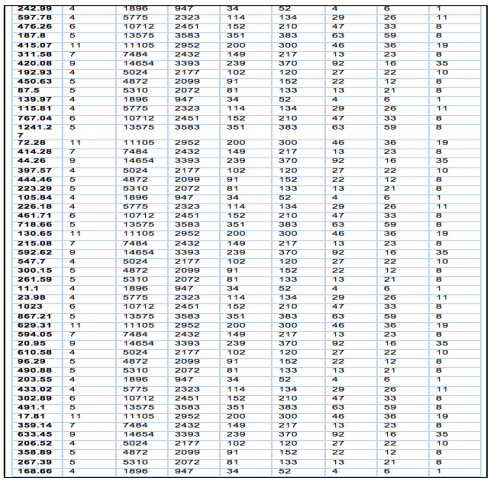
139.29	5	4872	2099	91	152	22	12	8
99	5	5310	2072	81	133	13	21	8
208.6	4	1896	947	34	52	4	6	1
82.88	4	5775	2323	114	134	29	26	11
96.92	6	10712	2451	152	210	47	33	8
181.92	5	13575	3583	351	383	63	59	8
305.64	11	11105	2952	200	300	46	36	19
680.07	7	7484	2432	149	217	13	23	8
992.91	9	14654	3393	239	370	92	16	35
306.75	4	5024	2177	102	120	27	22	10
83.9	5	4872	2099	91	152	22	12	8
481.04	5	5310	2072	81	133	13	21	8
113.93	4	1896	947	34	52	4	6	1
158.67	4	5775	2323	114	134	29	26	11
372.83	6	10712	2451	152	210	47	33	8
147.2	5	13575	3583	351	383	63	59	8
231.98	11	11105	2952	200	300	46	36	19
201.32	7	7484	2432	149	217	13	23	8
191.41	9	14654	3393	239	370	92	16	35
175.24	4	5024	2177	102	120	27	22	10
288.52	5	4872	2099	91	152	22	12	8
272.95	5	5310	2072	81	133	13	21	8
356.73	4	1896	947	34	52	4	6	1
391.86	4	5775	2323	114	134	29	26	11
271.35	6	10712	2451	152	210	47	33	8
169	5	13575	3583	351	383	63	59	8
540.4	11	11105	2952	200	300	46	36	19
184.93	7	7484	2432	149	217	13	23	8
297.67	9	14654	3393	239	370	92	16	35
274.91	4	5024	2177	102	120	27	22	10
641.58	5	4872	2099	91	152	22	12	8
859.63	5	5310	2072	81	133	13	21	8
139.19	4	1896	947	34	52	4	6	1
605.96	4	5775	2323	114	134	29	26	11
54.08	6	10712	2451	152	210	47	33	8
85.36	5	13575	3583	351	383	63	59	8
286.75	11	11105	2952	200	300	46	36	19
229.36	7	7484	2432	149	217	13	23	8
198.77	9	14654	3393	239	370	92	16	35
254.56	4	5024	2177	102	120	27	22	10
323.1	5	4872	2099	91	152	22	12	8
378.49	5	5310	2072	81	133	13	21	8
139.23	4	1896	947	34	52	4	6	1
132.71	4	5775	2323	114	134	29	26	11
205.77	6	10712	2451	152	210	47	33	8
176.96	5	13575	3583	351	383	63	59	8
399.62	11	11105	2952	200	300	46	36	19
777.69	7	7484	2432	149	217	13	23	8
1031.8	9	14654	3393	239	370	92	16	35
1								
254.11	4	5024	2177	102	120	27	22	10
147.96	5	4872	2099	91	152	22	12	8



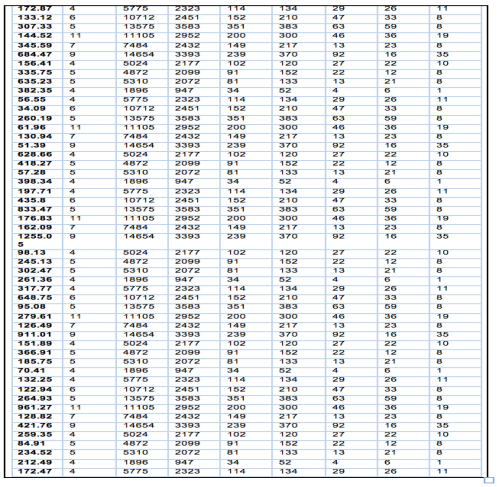
61.45	5	5310	2072	81	133	13	21	8
218.33	4	1896	947	34	52	4	6	1
124.81	4	5775	2323	114	134	29	26	11
1116.7	6	10712	2451	152	210	47	33	8
7								
633.98	5	13575	3583	351	383	63	59	8
112.16	11	11105	2952	200	300	46	36	19
147.76	7	7484	2432	149	217	13	23	8
304.72	9	14654	3393	239	370	92	16	35
230.78	4	5024	2177	102	120	27	22	10
380.46	5	4872	2099	91	152	22	12	8
60.65	5	5310	2072	81	133	13	21	8
37.88	4	1896	947	34	52	4	6	1
133.37	4	5775	2323	114	134	29	26	11
161.81	6	10712	2451	152	210	47	33	8
765.28	5	13575	3583	351	383	63	59	8
524.17	11	11105	2952	200	300	46	36	19
221.29	7	7484	2432	149	217	13	23	8
188.98	9	14654	3393	239	370	92	16	35
389.16	4	5024	2177	102	120	27	22	10
947.27	5	4872	2099	91	152	22	12	8
73.92	5	5310	2072	81	133	13	21	8
96.62	4	1896	947	34	52	4	6	1
103.14	4	5775	2323	114	134	29	26	11
753.12	6	10712	2451	152	210	47	33	8
594.36	5	13575	3583	351	383	63	59	8
596.45	11	11105	2952	200	300	46	36	19
509.17	7	7484	2432	149	217	13	23	8
377.73	9	14654	3393	239	370	92	16	35
154.35	4	5024	2177	102	120	27	22	10
290.24	5	4872	2099	91	152	22	12	8
165.04	5	5310	2072	81	133	13	21	8
106.36	4	1896	947	34	52	4	6	1
277.27	4	5775	2323	114	134	29	26	11
213.29	6	10712	2451	152	210	47	33	8
672.42	5	13575	3583	351	383	63	59	8
378.09	11	11105	2952	200	300	46	36	19
164.63	7	7484	2432	149	217	13	23	8
348.51	9	14654	3393	239	370	92	16	35
295.63	4	5024	2177	102	120	27	22	10
691.28	5	4872	2099	91	152	22	12	8
131.98	5	5310	2072	81	133	13	21	8
323.52	4	1896	947	34	52	4	6	1
769.36	4	5775	2323	114	134	29	26	11
179.48	6	10712	2451	152	210	47	33	8
504.68	5	13575	3583	351	383	63	59	8
286.19	11	11105	2952	200	300	46	36	19
131.4	7	7484	2432	149	217	13	23	8
489.34	9	14654	3393	239	370	92	16	35
106.8	4	5024	2177	102	120	27	22	10
828.26	5	4872	2099	91	152	22	12	8



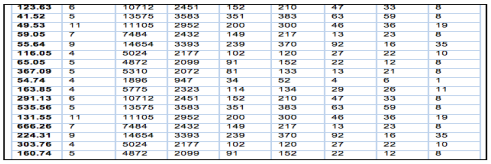
538.84	5	5310	2072	81	133	13	21	8
312.03	4	1896	947	34	52	4	6	1
462.38	4	5775	2323	114	134	29	26	11
190.37	6	10712	2451	152	210	47	33	8
283.5	5	13575	3583	351	383	63	59	8
459.71	11	11105	2952	200	300	46	36	19
243.1	7	7484	2432	149	217	13	23	8
182.14	9	14654	3393	239	370	92	16	35
557.27	4	5024	2177	102	120	27	22	10
264.77	5	4872	2099	91	152	22	12	8
185.02	5	5310	2072	81	133	13	21	8
13.92	4	1896	947	34	52	4	6	1
83.82	4	5775	2323	114	134	29	26	11
21.81	6	10712	2451	152	210	47	33	8
820.23	5	13575	3583	351	383	63	59	8
251.94	11	11105	2952	200	300	46	36	19
36.19	7	7484	2432	149	217	13	23	8
1576.4	9	14654	3393	239	370	92	16	35
2								
30.38	4	5024	2177	102	120	27	22	10
659.63	5	4872	2099	91	152	22	12	8
153.83	5	5310	2072	81	133	13	21	8
140.69	4	1896	947	34	52	4	6	1
131.51	4	5775	2323	114	134	29	26	11
249.66	6	10712	2451	152	210	47	33	8
617.98	5	13575	3583	351	383	63	59	8
1284.8	11	11105	2952	200	300	46	36	19
46.11	7	7484	2432	149	217	13	23	8
838.61	9	14654	3393	239	370	92	16	35
132.75	4	5024	2177	102	120	27	22	10
18.67	5	4872	2099	91	152	22	12	8
141.91	5	5310	2072	81	133	13	21	8
321.12	4	1896	947	34	52	4	6	1
157.83	4	5775	2323	114	134	29	26	11
149.1	6	10712	2451	152	210	47	33	8
686.16	5	13575	3583	351	383	63	59	8
131.11	11	11105	2952	200	300	46	36	19
299.23	7	7484	2432	149	217	13	23	8
270.59	9	14654	3393	239	370	92	16	35
130.05	4	5024	2177	102	120	27	22	10
654.09	5	4872	2099	91	152	22	12	8
494.78	5	5310	2072	81	133	13	21	8
23.58	4	1896	947	34	52	4	6	1
379.08	4	5775	2323	114	134	29	26	11
813.95	6	10712	2451	152	210	47	33	8
392.56	5	13575	3583	351	383	63	59	8
690.39	11	11105	2952	200	300	46	36	19
256.76	7	7484	2432	149	217	13	23	8
352.69	9	14654	3393	239	370	92	16	35
138.17	4	5024	2177	102	120	27	22	10
415.2	5	4872	2099	91	152	22	12	8
270.98	5	5310	2072	81	133	13	21	8



242.99	4	1896	947	34	52	4	6	1
597.78	4	5775	2323	114	134	29	26	11
476.26	6	10712	2451	152	210	47	33	8
187.8	5	13575	3583	351	383	63	59	8
415.07	11	11105	2952	200	300	46	36	19
311.58	7	7484	2432	149	217	13	23	8
420.08	9	14654	3393	239	370	92	16	35
192.93	4	5024	2177	102	120	27	22	10
450.63	5	4872	2099	91	152	22	12	8
87.5	5	5310	2072	81	133	13	21	8
139.97	4	1896	947	34	52	4	6	1
115.81	4	5775	2323	114	134	29	26	11
767.04	6	10712	2451	152	210	47	33	8
1241.2	5	13575	3583	351	383	63	59	8
7								
72.28	11	11105	2952	200	300	46	36	19
414.28	7	7484	2432	149	217	13	23	8
44.26	9	14654	3393	239	370	92	16	35
397.57	4	5024	2177	102	120	27	22	10
444.46	5	4872	2099	91	152	22	12	8
223.29	5	5310	2072	81	133	13	21	8
105.84	4	1896	947	34	52	4	6	1
226.18	4	5775	2323	114	134	29	26	11
461.71	6	10712	2451	152	210	47	33	8
718.66	5	13575	3583	351	383	63	59	8
130.65	11	11105	2952	200	300	46	36	19
215.08	7	7484	2432	149	217	13	23	8
592.62	9	14654	3393	239	370	92	16	35
547.7	4	5024	2177	102	120	27	22	10
300.15	5	4872	2099	91	152	22	12	8
261.59	5	5310	2072	81	133	13	21	8
11.1	4	1896	947	34	52	4	6	1
23.98	4	5775	2323	114	134	29	26	11
1023	6	10712	2451	152	210	47	33	8
867.21	5	13575	3583	351	383	63	59	8
629.31	11	11105	2952	200	300	46	36	19
594.05	7	7484	2432	149	217	13	23	8
20.95	9	14654	3393	239	370	92	16	35
610.58	4	5024	2177	102	120	27	22	10
96.29	5	4872	2099	91	152	22	12	8
490.88	5	5310	2072	81	133	13	21	8
203.55	4	1896	947	34	52	4	6	1
433.02	4	5775	2323	114	134	29	26	11
302.89	6	10712	2451	152	210	47	33	8
491.1	5	13575	3583	351	383	63	59	8
17.81	11	11105	2952	200	300	46	36	19
359.14	7	7484	2432	149	217	13	23	8
633.45	9	14654	3393	239	370	92	16	35
206.52	4	5024	2177	102	120	27	22	10
358.89	5	4872	2099	91	152	22	12	8
267.39	5	5310	2072	81	133	13	21	8
168.66	4	1896	947	34	52	4	6	1



172.87	4	5775	2323	114	134	29	26	11
133.12	6	10712	2451	152	210	47	33	8
307.33	5	13575	3583	351	383	63	59	8
144.52	11	11105	2952	200	300	46	36	19
345.59	7	7484	2432	149	217	13	23	8
684.47	9	14654	3393	239	370	92	16	35
156.41	4	5024	2177	102	120	27	22	10
335.75	5	4872	2099	91	152	22	12	8
635.23	5	5310	2072	81	133	13	21	8
382.35	4	1896	947	34	52	4	6	1
56.85	4	5775	2323	114	134	29	26	11
34.09	6	10712	2451	152	210	47	33	8
260.19	5	13575	3583	351	383	63	59	8
61.96	11	11105	2952	200	300	46	36	19
130.94	7	7484	2432	149	217	13	23	8
51.39	9	14654	3393	239	370	92	16	35
628.66	4	5024	2177	102	120	27	22	10
418.27	5	4872	2099	91	152	22	12	8
57.28	5	5310	2072	81	133	13	21	8
398.34	4	1896	947	34	52	4	6	1
197.71	4	5775	2323	114	134	29	26	11
435.8	6	10712	2451	152	210	47	33	8
833.47	5	13575	3583	351	383	63	59	8
176.83	11	11105	2952	200	300	46	36	19
162.09	7	7484	2432	149	217	13	23	8
1255.0	9	14654	3393	239	370	92	16	35
5								
98.13	4	5024	2177	102	120	27	22	10
245.13	5	4872	2099	91	152	22	12	8
302.47	5	5310	2072	81	133	13	21	8
261.36	4	1896	947	34	52	4	6	1
317.77	4	5775	2323	114	134	29	26	11
648.75	6	10712	2451	152	210	47	33	8
95.08	5	13575	3583	351	383	63	59	8
279.61	11	11105	2952	200	300	46	36	19
126.49	7	7484	2432	149	217	13	23	8
911.01	9	14654	3393	239	370	92	16	35
151.89	4	5024	2177	102	120	27	22	10
366.91	5	4872	2099	91	152	22	12	8
185.75	5	5310	2072	81	133	13	21	8
70.41	4	1896	947	34	52	4	6	1
132.25	4	5775	2323	114	134	29	26	11
122.94	6	10712	2451	152	210	47	33	8
264.93	5	13575	3583	351	383	63	59	8
961.27	11	11105	2952	200	300	46	36	19
128.82	7	7484	2432	149	217	13	23	8
421.76	9	14654	3393	239	370	92	16	35
259.35	4	5024	2177	102	120	27	22	10
84.91	5	4872	2099	91	152	22	12	8
234.52	5	5310	2072	81	133	13	21	8
212.49	4	1896	947	34	52	4	6	1
172.47	4	5775	2323	114	134	29	26	11



123.63	6	10712	2451	152	210	47	33	8
41.52	5	13575	3583	351	383	63	59	8
49.53	11	11105	2952	200	300	46	36	19
59.05	7	7484	2432	149	217	13	23	8
55.64	9	14654	3393	239	370	92	16	35
116.05	4	5024	2177	102	120	27	22	10
65.05	5	4872	2099	91	152	22	12	8
367.09	5	5310	2072	81	133	13	21	8
54.74	4	1896	947	34	52	4	6	1
163.85	4	5775	2323	114	134	29	26	11
291.13	6	10712	2451	152	210	47	33	8
535.56	5	13575	3583	351	383	63	59	8
131.55	11	11105	2952	200	300	46	36	19
666.26	7	7484	2432	149	217	13	23	8
224.31	9	14654	3393	239	370	92	16	35
303.76	4	5024	2177	102	120	27	22	10
160.74	5	4872	2099	91	152	22	12	8

Figure E.5: The time taken and variation of metrics for all the 32 participants are displayed.

Pilot Study 1:

The time taken by each participant to answer the questions and the variation in the metrics for the question that is answered by the participant are addressed below. This will allow to perform the regression analysis on the data points to understand which metric is a good predictor over human oracle costs.

Q.ID	Time	Depth	Size	Compress	Tags
1	13.52	5	5310	2072	81
3	147.38	4	1896	947	34
6	252.13	4	5775	2323	114
7	191.86	6	10712	2451	152
9	529.89	5	13575	3583	351
11	170.47	11	11105	2952	200
13	347.32	7	7484	2432	149
16	28.08	9	14654	3393	239
17	241.84	4	5024	2177	102
19	140.54	5	4872	2099	91
1	369.52	5	5310	2072	81
3	1000	4	1896	947	34
6	274.34	4	5775	2323	114
7	465.84	6	10712	2451	152
9	392.71	5	13575	3583	351
11	152.84	11	11105	2952	200
13	525.72	7	7484	2432	149
16	38.19	9	14654	3393	239
17	336.4	4	5024	2177	102
19	51.91	5	4872	2099	91
1	15.09	5	5310	2072	81
3	783.7	4	1896	947	34
6	11.5	4	5775	2323	114
7	14.53	6	10712	2451	152
9	11.78	5	13575	3583	351
11	13.57	11	11105	2952	200
13	11.8	7	7484	2432	149
16	46.61	9	14654	3393	239
17	63.47	4	5024	2177	102
19	1000	5	4872	2099	91
1	1000	5	5310	2072	81
3	14.34	4	1896	947	34
6	522.96	4	5775	2323	114
7	19.19	6	10712	2451	152
9	19.65	5	13575	3583	351
11	14.79	11	11105	2952	200
13	35.16	7	7484	2432	149
16	23.34	9	14654	3393	239
17	20	4	5024	2177	102
19	15	5	4872	2099	91

Table E.5: The results from all the four participants illustrating how much time they have taken to attempt each test input; time is in seconds unit.

Results from pilot study 2

For the Pilot Study 2 we present all four metrics for every test input along with the

variation in the metrics for all the questions, the time taken to answer each test input by every participant is given in the table below. This allows to perform regression analysis on the data points to understand which metric is a good predictor of human oracle costs.

Question id	**time**	**size**	**compress**	**depth**	**tags**
2	45.1	5310	2072	81	4
4	78.15	1905	931	34	4
5	263.99	5921	1695	124	8
8	201.11	9856	2247	143	5
10	258.31	8532	2576	158	5
12	317.97	10186	2513	200	9
14	166.55	4464	1647	99	3
15	811.48	14654	3386	239	7
18	62.74	5024	2237	102	4
19	298.39	4464	1647	99	3
2	429.74	5310	2072	81	4
4	126.54	1905	931	34	4
5	82.63	5921	1695	124	8
8	337.77	9856	2247	143	5
10	300.85	8532	2576	158	5
12	252.8	10186	2513	200	9
14	15.88	4464	1647	99	3
15	277.78	14654	3386	239	7
18	370.49	5024	2237	102	4
19	345.9	4464	1647	99	3
2	70.19	5310	2072	81	4
4	118.24	1905	931	34	4
5	155.69	5921	1695	124	8
8	678.12	9856	2247	143	5
10	97.13	8532	2576	158	5
12	100.51	10186	2513	200	9
14	208.05	4464	1647	99	3
15	115.6	14654	3386	239	7
18	393.46	5024	2237	102	4
19	294	4464	1647	99	3
2	151.56	5310	2072	81	4
4	324.83	1905	931	34	4
5	84.91	5921	1695	124	8
8	175.12	9856	2247	143	5
10	95.13	8532	2576	158	5
12	680.79	10186	2513	200	9
14	140.35	4464	1647	99	3
15	177.34	14654	3386	239	7
18	234.2	5024	2237	102	4
19	505.94	4464	1647	99	3

Table E.6: The results show time participants have taken to attempt each test input

Table representing the size and the compress size:
Important note here is to look at all the 4 metrics distribution of the corresponding HTML example. These metrics are used as part of 2 Pilot studies and the Experiment.

Test Input	**Size of the entire folder**	**Compress size of the entire folder**	**Size of the HTML test input**	**Compress size of the HTML test input**
Art Gallery	28 63 151	28,50,820	5310	2072
Black coffee	3 30 293	2,75,319	5921	1695
Blue media	1 30 012	97,785	10712	2,451
Blue simple template	2 20 309	1,06,557	13575	3,583
Cooperation	82,810	36,945	8532	2,576
Templated coefficient	6,23,681	4,06,914	5024	2,237
Templated Intensity	15,11,078	9,05,146	4,872	2,151
Templated lady tulip	5,28,367	3,08,615	5,775	2,323
Studio	34,02,558	27,90,421	14,654	3,395
HTML 5 up Ariel	13,47,500	8,64,645	1,896	947
HTML 5 up Escape Velocity	15,34,679	9,45,676	11,105	2,952
Forty	20,65,651	14,80,746	7,484	2,432

Table E.7: The Metrics size, compress size of each HTML test input both at the entire folder level and individual index.html

START SET ARTICLES	
1	F. Pastore, L. Mariani, and G. Fraser, "CrowdOracles: Can the Crowd Solve the Oracle Problem?," in *Verification and Validation 2013 IEEE Sixth International Conference on Software Testing*, 2013, pp. 342–351.
2	S. A. Ajila and R. T. Dumitrescu, "Experimental use of code delta, code churn, and rate of change to understand software product line evolution," *J. Syst. Softw.*, vol. 80, no. 1, pp. 74–91, Jan. 2007.
3	R. Feldt and S. Poulding, "Finding test data with specific properties via metaheuristic search," in *2013 IEEE 24th International Symposium on Software Reliability Engineering (ISSRE)*, 2013, pp. 350–359.
4	R. N. Zaeem, M. R. Prasad, and S. Khurshid, "Automated Generation of Oracles for Testing User-Interaction Features of Mobile Apps," in *Verification and Validation 2014 IEEE Seventh International Conference on Software Testing*, 2014, pp. 183–192.
5	S. R. Dalal and A. A. McIntosh, "When to stop testing for large software systems with changing code," *IEEE Trans. Softw. Eng.*, vol. 20, no. 4, pp. 318–323, Apr. 1994.
6	P. McMinn, M. Stevenson, and M. Harman, "Reducing qualitative human oracle costs associated with automatically generated test data," in *Proceedings of the First International Workshop on Software Test Output Validation*, 2010, pp. 1–4.
7	S. Afshan, P. McMinn, and M. Stevenson, "Evolving Readable String Test Inputs Using a Natural Language Model to Reduce Human Oracle Cost," in *Verification and Validation 2013 IEEE Sixth International Conference on Software Testing*, 2013, pp. 352–361.
8	G. Manduchi and C. Taliercio, "Measuring software evolution at a nuclear fusion experiment site: a test case for the applicability of OO and reuse metrics in software characterization," *Inf. Softw. Technol.*, vol. 44, no. 10, pp. 593–600, Jul. 2002.
9	E. T. Barr, M. Harman, P. McMinn, M. Shahbaz, and S. Yoo, "The Oracle Problem in Software Testing: A Survey," *IEEE Trans. Softw. Eng.*, vol. 41, no. 5, pp. 507–525, May 2015.
10	S. Poulding and R. Feldt, "The automated generation of humancomprehensible XML test sets," in *Proc. 1st North American Search Based Software Engineering Symposium (NasBASE)*, 2015.
11	T. Kanstren, "Program Comprehension for User-Assisted Test Oracle Generation," in *2009 Fourth International Conference on Software Engineering Advances*, 2009, pp. 118–127.

Figure E.6: Start Set Articles