
Evaluating Solution Correctness and Problem Difficulty Based on Code Metrics

Gurpreet Singh (# 250674134)
The University Of Western Ontario

GSINGH95@UWO.CA

Abstract

NOTE this is very important stuff!!
this is an example of an abstract.this is an example of an abstract.this is an example of an abstract.this is an example of an abstract.this is an example of an abstract.this is an example of an abstract.this is an example of an abstract.this is an example of an abstract.this is an example of an abstract.this is an example of an abstract.

1. Description of Applied Problem

1.1. Problem

A large amount of educational material related to programming exists on the internet but the majority of which is not well structured or presented. An applied problem that can be observed from educational material found online is that code quality is often left mutually exclusive from code functionality. [Astrachan \(2004\)](#)

This leads to some readers believing it is acceptable to write code that produces the correct result even if the process behind it is not correct. Online code challenge websites like CodeChef.com do not take into account the style and quality metrics of a code submission when judging competitions. [CodeChef \(2017\)](#) Cutting corners in the learning process advances into a complete disregard for best practices in open source software and in the workplace which results in a larger amount of errors. Readability of code is an essential metric in software engineering and can be improved even with simple additions of whitespace between lines. [Buse and Weimer \(2010\)](#)

Computer code written by researchers and other indi-

viduals who are just trying to accomplish a result in any way possible is often of the worse quality. This is because they learn using the ‘just in time’ mentality and the resources online that promote this mentality disregard code quality. [Astrachan \(2004\)](#) Lack of code quality directly reduces it’s re-usability because other programmers have a harder time understanding what the code is doing. If these teaching resources could use questions and checks that promote better code quality, many technical innovations could be made. Researchers would be more educated on how to create reusable code and this would influence developers to take their ideas and apply them to real life use cases. [Gandhi and Bhatia \(2010\)](#)

Code quality post processing software is often used in production development environments to ensure good style choices. These checks are much less useful at this senior level than they would be at an educational level. If programming style can be judged on a submission, companies conducting technical interviews will be able to better judge applicants and make a more informed decision.

This study will focus on proving that code quality can have an influence on code functionality, as well as which kinds of questions influence good or bad code styles.

1.2. Proposed Solution

A solution to these problems is linking the scoring process in programming problems to a metric derived from running code quality checks on the submission.

Not only will this analysis benefit educational institutes but also companies and competitions that judge people on their code submissions.

2. Description of Available Data

2.1. CodeChef Dataset

CodeChef.com is a competitive programming web application that has posted all of their questions and solutions onto the data science website, Kaggle. The data consists of a questions comma separated file, a solutions comma separated file and 3 files that show the code associated with each solution id. The set contains about 1000 problem statements and over 1 million code solutions submitted. This should be a more than sufficient to make a training and test data set.

The important features available for each question are:

- title
- link
- difficulty level
- question statement
- time limit

The important features available for each solution are:

- status (correct or wrong)
- time taken
- memory taken
- language written in
- solution url

2.2. Filtering by popular languages

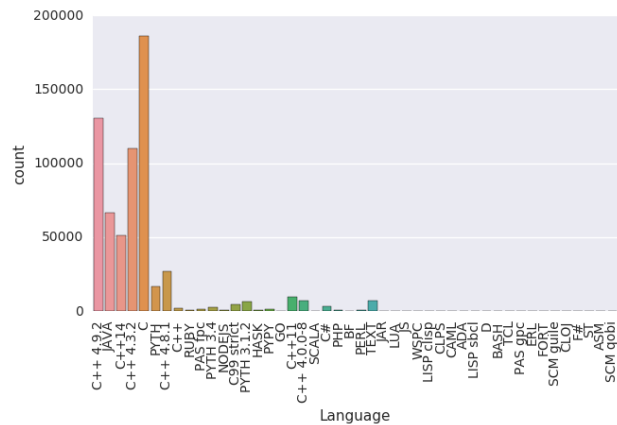
The code submissions are written in many different programming languages and each language has its own code analysis tool. Therefore, to make the process simpler and come up with higher quality results, the data will need to be filtered by the top languages used. Figure 1 shows that C++, Java, C and Python are the most popular submissions in this dataset. There are 4 versions of C++ but it should be possible to process them with one tool.

Wang (2016)

3. Plan for Analysis and Visualization

3.1. Description

In order to make the connection between code functionality and code style, a large part of the experiment



220	approach. <i>International Journal of Software Engi-</i>	275
221	<i>neering (IJSE)</i> , 1(4):63–72, 2010.	276
222		277
223	Justin Wang. Nlp and ml experiments, December	278
224	2016. URL https://www.kaggle.com/justwjr/	279
225	nlp-and-ml-experiments/notebook . [Accessed	280
226	Oct 26, 2017].	281
227		282
228		283
229		284
230		285
231		286
232		287
233		288
234		289
235		290
236		291
237		292
238		293
239		294
240		295
241		296
242		297
243		298
244		299
245		300
246		301
247		302
248		303
249		304
250		305
251		306
252		307
253		308
254		309
255		310
256		311
257		312
258		313
259		314
260		315
261		316
262		317
263		318
264		319
265		320
266		321
267		322
268		323
269		324
270		325
271		326
272		327
273		328
274		329