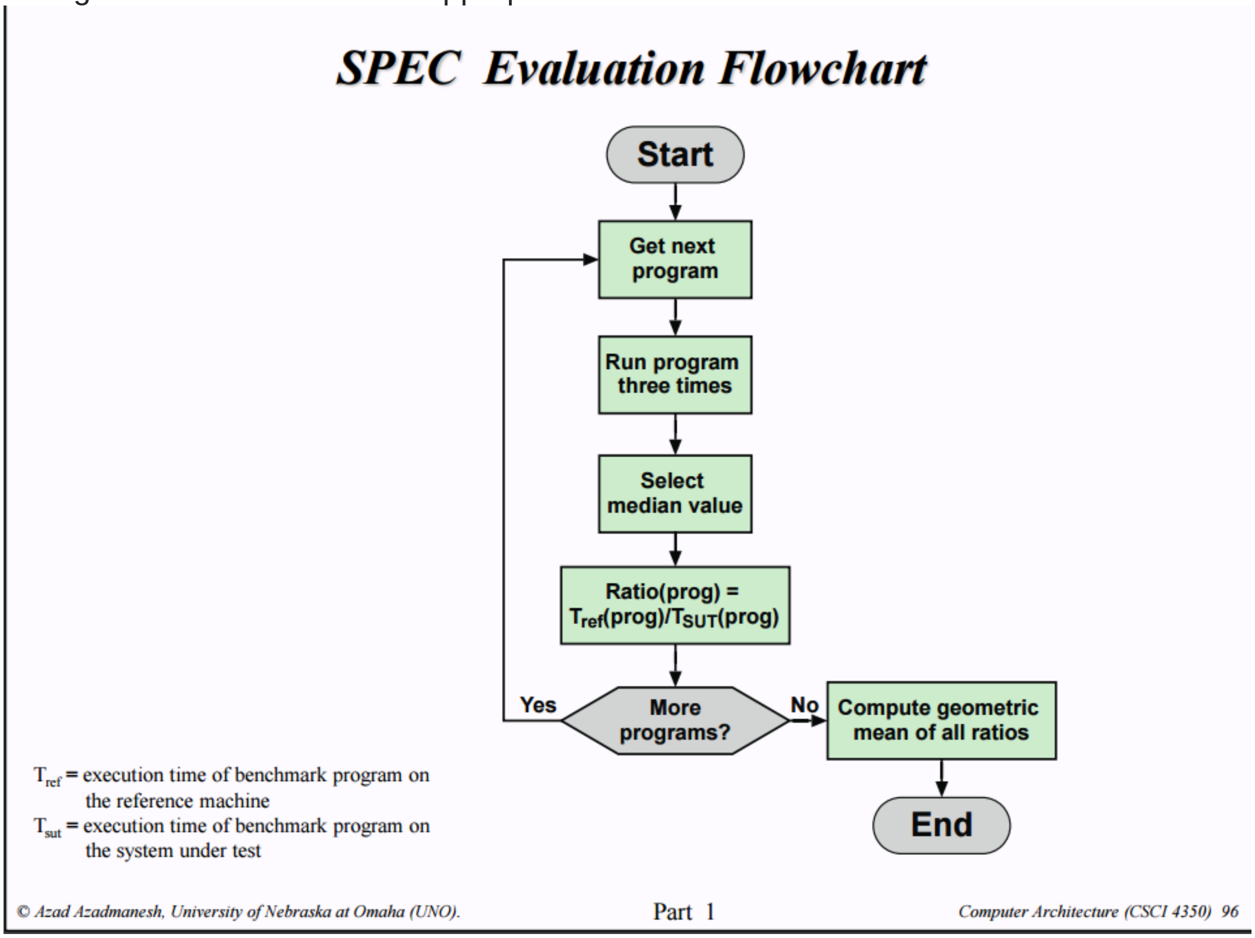
Solution proposal

By Kiet Tran

Compare ScanCode runtimes with Fossology, licensee, LicenseFinder, license-check, ninka, slic, LiD and others. This project is to create a comprehensive test suite and a benchmark for several FOSS open source license and copyright detection engines, establish mappings between the different conventions they use for license identification and evaluate and publish the results of detection accuracy and precision.

1/ Compare the speed of scanning

In order to know which license scanner runs faster, we need a set of data and formula to calculate the result for each. I came across with this formula while studying computer architecture this semester and I found this fascinating. SPEC CPU 2006 benchmark This can be used to calculate the speed of scanning of these license scanners. We will change it a little bit to make it appropriate for this



SPEC Evaluation Flowchart

T_{ref} = execution time of benchmark program on the reference machine

T_{sut} = execution time of benchmark program on the system under test

© Azad Azadmanesh, University of Nebraska at Omaha (UNO). Part 1 *Computer Architecture (CSCI 4350) 96*

This image above is created by Dr Azad Azadmanesh, professor at University of Omaha Nebraska

Basically, we will have roughly around 5 data with the same property to scan (personally, I think 5 is a good number, not too big or small) Each program will scan the file 3 times, select the median value and continue to scan other exactly the same. After that, we will compute geometric mean of these median and we will have the speed for that software. After that, we will draw a bar chart to show the result. We can enhance this by testing with different kind of data. Every scanner has strong and weak points, therefore, it will be much better if we test with different kind of data. The question is how to get data and how to measure the scanning time. Perhaps, we will write a python script file to measure this.

2/ Compare the accuracy

This is more challenge. There are two possible cases for this. Data with known results and data with unknown results.

Data with known results:

If we have already known exactly how many licenses in the file, all we need to do is just to compare the result and the data to detect the differences. Based on that, we will calculate its accuracy. Of course, wrong license can be found. Those wrong licenses should be left aside and we will recalculate later to get better accuracy point. For example, there are 100 license and license scanner only detect 40, 5 of them are wrong information, the accuracy will be 35%. These 5 extra licenses (which cannot be found in original data) should be considered to see if they are important or not. After that, we will recalculate the accuracy.

Data with unknown results:

Imagine that we need to calculate the accuracy of 5 license scanners and we have 1 folder and we do not know exactly how many licenses in that folder. For example: those scanners will output 50 60 65 60 70. Firstly, we get the one with the highest number. If possible, we should check manually to make sure that all the licenses that are detected by the scanner with highest number are correct. Secondly, we need to compare the difference between that and another license scanner to update our license in the folder. If it detects something that we forget to check during the first process, we will update it. Keep doing that with the remaining and calculate its accuracy

Conclusion:

Finding the accuracy is not an easy task. If we have good clean data to test with, thing will be easy. If not, we need to check the result from the scanners at first, and then we will update our data when we go on

About the author:

I’m currently undergrad student at University of Omaha Nebraska and major in Computer Science. My interest is Software development and machine learning. I’m looking for an internship to improve my coding skills. I mainly work with Java and Python. If you have any questions about my proposal, please email me at khtran@unomaha.edu